# PROGRESSIVE DATA MINING: AN EXPLORATION OF USING WHOLE-DATASET FEATURE SELECTION IN BUILDING CLASSIFIERS ON THREE BIOLOGICAL PROBLEMS

By
SUNDARARAJAN VIJAYARAGHAVA SESHADRI

A THESIS SUBMITTED
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT
NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING
3 Science Drive 2, Singapore 117543
ATTACHED TO
INSTITUTE FOR INFOCOMM RESEARCH
21 Heng Mui Keng Terrace, Singapore 119613.

*To Mataji*

# ACKNOWLEDGMENT

Prof. Limsoon Wong, Professor, SOC, NUS (former Research Director, Institute for Infocomm Research) should be remembered even before opening this thesis report. His continuous encouragement from the beginning gave me full energy and enthusiasm in achieving this Ph.D degree under him at NUS.

Prof. See-Kiong Ng, Department Manager, Knowledge Discovery Department (KDD), Institute for Infocomm Research, suggested to me a wonderful project on function prediction for the yeast genome, when I was searching for a topic. Even though the project was quite tough, his boosting ideas made me to eventually solve specific classification problems in handling multiple data sets.

Mr. Soon-Heng Tan, Biologist at KDD, Institute for Infocomm Research, was a day-to-day tonic to me in knowing biological insights to the experiments and the data sets that we have downloaded from Stanford Microarray Database.

Prof. Anthony K.H. Tung, SOC, NUS, taught me "Knowledge Discovery in Databases" which inspired me to take a research project in that domain. Prof. David Hsu, SOC, NUS, taught me "Motion Planning and Applications", which eventually inspired me to take a project in modeling types of protein sites. Prof. Jinyan Li, Institute for Infocomm Research, gave very useful ideas on research problems. Prof Wing Kin Sung was first known to me when I attended "Combinatorial methods in bioinformatics" at SOC, NUS. Later, he suggested many useful issues on my thesis. Dr. Huiqing Liu helped me understand the WEKA package and always addressed issues with a smile. Dr. Haiquan Li regularly guided me on issues in my thesis. Judice Koh, Donny Soh and many others at my lab shared lots of suggestions and knowledge. I sincerely thank Institute for Infocomm Research in funding my scholarship, a conference trip, computer systems, and other day-to-day requirements in the lab.

Last but not least, I love to thank my wife, Subasri, who made huge sacrifices and contributions, in making this Ph.D thesis possible mentally and physically.

# ABSTRACT

MOTIVATION : Building efficient classification model using limited data is a challenging problem. Each microarray experiment provides information about the behavior of possibly a large number of genes, but only within the specific experimental setup. So, the behavior of the same gene set is not known for different cell conditions. Each data set from laboratory experiments can be used to mine rich associative information regarding involved genes from other resources, so that much more information can be derived than what the original experiment provides for. One of the important questions in general genomics and proteomics is elucidation of the function of proteins and how to determine these from the available data. Generally, proteins perform their function in cells by interacting with other molecules. Thus, determining their binding environments is very important. These interaction protein segments are generally known as protein active sites. Once we have derived the biochemical properties or micro-environment properties surrounding an active protein site, we can use these to build models for recognition of different types of these sites. In a broader context, some of the protein functions are reflected in the different protein characteristics. Machine learning methods are useful to build prediction and classification models for these purposes. For example, previously applied methods for recognition of protein active sites include Naïve Bayesian algorithm to predict calcium binding sites from structural properties surrounding these sites. Also, some of the previous studies in *S. cerevisiae* genes attempted to predict 96 gene functions using multilayer perceptron and outcomes of only six microarray experiments, but results have shown that only 10% of functions could be predicted by that approach. This implies that generation of good classification models may not be feasible with limited biological data.

PROBLEM DEFINITION : Previous studies on recognition of protein active sites used a rich collection of various features for creating their recognition models. These features have been generally classified into several functional groups. The abovemention studies used the whole set of these features without investigating the issue of the optimal choice of feature combinations or the combination of functional groups

of features. The studies of protein functions based on limited microarray experiments have shown that much richer data sources are required while the optimized selection of the features in this context has not been considered. In view of this we address a research problem described as "Progressive Data Mining: An Exploration of Using Whole-Dataset Feature Selection in Building Classifiers on Three Biological Problems" that develops specific method of optimized feature selection and illustrates the results on three specific problems. These problems are a) recognition of five functions of yeast genes based on features selected from six micorarray datasets; b) recognition of three types of protein active sites based on six categories of micro-environment properties; c) modeling of 46 protein functions in yeast based on 57 microarray experiments.

CONTRIBUTION : Our research focuses on selecting the most useful sub-set of data from the given dataset in achieving a higher recognition performances of models built on these data than what can be achieved by the conventional methods. Specifically:

1. We proposed "Hill-climbing algorithm" and "Greedy-Hill climbing algorithm" to select features to enhance performance of classification models. Progressive data-mining, Hill-based, and Greedy-Hill-based algorithms for feature selection and for selection of combination of feature groups.

2. We demonstrate by the comparison results of different methods used that the conventional methods (based on the best feature data set, all available data sets, and features selected by conventional feature selection methods) perform poorer to those based on the Hill and Greedy-Hill feature selection methods.

3. We also demonstrate that the progressive data mining concept improves performance of generated classifiers, as well as that the combination of the whole data sets selected by Hill or Greedy-Hill algorithms results in better classification models than the conventional feature selection algorithms. We demonstrated a better classification performance (by eight evaluation metrics) by Hill-based feature selection method than by the conventional methods on three biological problems.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

Microarray technology allows researchers to conduct experiment and monitor expression of many genes simultaneously, over different time points. Each experiment may have a specific objective in studying a subset of genes in a particular functional pathway. For example, Fernandes *et al* [11] studied yeast genes by setting an experimental condition "high hydrostatic pressure" to identify 274 genes belonging to "stress response" function.

Though each such experiment focuses on a subset of genes of a specific function or under a specific experimental condition, the experiments reveal the gene expression profiles of all genes. Thousands of gene expression profiles have been recorded in the literature. Little studies have tried to utilize those datasets to generate more knowledge about the genes. For instance, the literature has many microarray expression datasets related to *S. cerevisiae* (yeast). Yet only 50% of yeast genes are currently functionally annotated.

Proteins interact by binding to each other to form complexes used by the molecular machinery of living cells to affect various biological processes. It is thus of increased importance to be able to analyze potential of proteins to bind to other proteins. These

interactions occur via protein binding sites. While experimental determination of the protein binding sites is preferable, computational approaches are more convenient as they are fast and inexpensive. For this reason the search for efficient computational methods to recognize protein binding sites is in high demand. In the computational recognition of protein binding sites many features have been proposed. Steven et al [57] formulated and characterized micro-environment properties surrounding protein binding sites. These micro-environment properties are based on the physical, chemical and structural characteristics that can be calculated from the 'atoms' [57] that provide information about protein residues and their neighborhood. Steven et al [57] showed that the distributions of micro-environment properties significantly differ between sites and 'non-sites'. This fact makes micro-environment properties useful as features for machine learning type recognitions of protein binding sites. Knowledge-based approach can overcome deficiencies in the current understanding of molecular recognition of the biological systems [55].

Biological experiments do have limitations in revealing associative knowledge that could be derived from primary findings. This phenomena motivates biologists to seek help from computational techniques. For example, Mateos *et al.* [34] went further to study the 5 cellular functions and tried to predict 96 functions of *S. cerevisiae* genes using neural networks. They reported that only 10% of functions are trainable by their approach—this is not surprising since many of the 96 functional classes have too few members or have ambiguous members. This shows that classification models based on limited biological knowledge is not useful. Current studies are conducted either by deriving clusters with data in an unsupervised manner or building classification models with known annotations as class labels with data.

## 1.1 Problem Statement

In our research, we study the optimal choice of using selected sub-sets of data in building efficient classification models. In this section we raise questions that are common among problems that possess multiple data sets.

We are using a Banking situation to make an analogy. **Banking** possesses voluminous of information from each customer. The information can be segmented into personal, economical, social, and educational categories. At the outset, these multiple sets of data do not show whether any applicant is a potential customer or not to the bank. Decision on an applicant is not made based on that particular person's data only. Then how does the Bank take a decision on *sanctioning a loan* or *analysing consumer behavior*? The bank's objective is to do the sanctioning only to genuine good customers. This raises some natural questions that are considered in the next subsection.

### 1.1.1 General Research Objective on Huge Amount of Data

1. Does limited information on a customer help in decision making?
2. Does additional data help in better decision making?
3. Does using all available data give the best decision?
4. Does applying filtering method help in better decision?
5. Does choosing important data help in efficient decision making?

Let us see how similar objectives are enlisted from the biological point of view in the next subsection.

### 1.1.2 Biological Research Objective on Multi Dimensional Data

Similar to the case study of "Bank", if we go through the available information on genes of a genome, we realise that only a limited amount of knowledge is so far uncovered through biological experiments. For example, a simple organism like *S. cerevisiae* has about 6400 genes; but only 50% of the genes are known functionally. Such limited knowledge is not sufficient to understand the complete cellular pathways of those genes. Understanding a complete genome helps in knowing functional aspects of proteins, protein structures and finally leads to design of drugs. On the other hand, computational techniques are capable of building a knowledge base with available limited data on known genes and use that knowledge for understanding unknown genes. This gives rise to useful and important research questions :

1. Can we use limited biological samples to build a proper classifier?
2. Can we use additional data sets of different experimental nature, on the same set of genes to improve a classifier?
3. Can we use all available data sets to achieve a better classifier?
4. Can we use selected features from conventional feature selection method to achieve a better classifier?
5. Can we combine selected data sets to yield a better classifier?

To address the research questions above and evaluate different classification methods, we focus our work on 1) use of individual data set or category or experiment; 2) selection of useful data sets for building accurate classification models; 3) achieve

better performance on classifiers with selected categories or data sub-sets. In particular, we propose "Progressive Data Mining (PDM)" to achieve higher performance with the combination of whole data sets through Hill and Greedy-Hill algorithms. We compare the results of study with other conventional approaches—using the best of individual data sets, using all available data sets, and using selected features by feature selection methods. We also evaluate each method by comparing their results with the optimum result using the combination of whole data sets through exhaustive search. We selected 3 bioinformatics problems in our research studies—5 specific functions of yeast genes, 3 types of protein sites, and 26 specific functions of yeast genes.

## 1.2 Introduction to our Research Studies

We address a research problem "Progressive Data Mining: An Exploration of Using Whole-Dataset Feature Selection in Building Classifiers on Three Biological Problem".

We choose the following three problems that were recently studied by using all available datasets. Researchers Brown *et al.*, Mateos *et al.*, Bagley *et al.* and Wei *et al.* used all available data sets in their classification models. They did not investigate the issue of the optimal choice of combinations of data sets. Using as small as possible set of features for building efficient models would be beneficial for the reason that not all features are always available for the models of interest.

In this section we introduce the three bioinformatics problems that are studied.

- The first problem we considered is on 5 functions of yeast genes which was earlier studied by Brown *et al.* [5] and Mateos *et al.* [34]. Wet experiments are

regularly conducted on a genome to derive a gene expression data set. Data sets of *S. cerevisiae* are the recent focus of many computer scientists due to its rich known annotations. Annotations are used with either individual or combined data sets in classification studies. Brown *et al.* [5] and Mateos *et al.* [34] studied functional classification problem on *S. cerevisiae* by using all available data sets.

- The second problem we considered is on 3 types of protein sites. Recognizing binding sites in a three-dimensional protein structure is necessary to fully appreciate the functional aspect of the protein. Identifying binding regions of a protein from structural characteristics by biological experiments are not easy. Recently, [31] used Naïve Bayesian algorithm to predict calcium binding sites from structural properties surrounding these sites.

- The third problem we considered is on 26 specific functions of yeast genes. Mateos *et al.* [34] studied 96 functions of yeast genes through multilayer perceptrons and reported that only 10% of functions were trainable by using gene expression data sets. Based on previous studies [5, 34, 8] we initially had 116 functions. We finally considered only 26 functions that involve more than 25 genes.

### 1.2.1 5 Specific Functions of Yeast Genes

It is not uncommon that microarray experiments on identical or similar sets of genes are repeatedly conducted by various laboratories for different functional studies of these genes. As such, multiple sets of microarray data on the same set of genes can often be collected from different laboratories and research centers, either through collaborators or from online gene expression data repositories. It will be useful if we can

effectively combine these additional diverse data sets (on same set of genes) with the data generated in one's laboratory to further improve our microarray data mining results. Although many of the microarray experiments may have been conducted on identical sets of genes, the studies are often designed to address different scientific questions, and are usually conducted under varying experimental conditions. For example, one microarray experiment may focus on identifying new components in polyphosphate metabolism using the gene knockout method [40], while another similar microarray experiment on the same set of genes may be designed to study spore morphogenesis [7]. Intuitively, it should be beneficial to combine the two gene expression data sets for microarray data analysis, given that they have been conducted on the same set of genes (both cited experiments used the *Saccharomyces cerevisiae*'s genome in their investigations). On the other hand, their differences in study objectives and experimental conditions may not warrant that combining or merging data sets based on same gene from these two different studies can improve data mining results. Modeling the functional aspect of genes is important in understanding the complete genomic activity of an organism. Biologists are interested in getting more and more accurate computational models with existing biological knowledge on functional annotations of genes. Studies on microarray experimental assays are becoming important for the functional classification of genes. Brown studied 5 functions of yeast genes with 6 data sets of yeast by learning algorithms [5]. Mateos conducted a similar study and extended it to 96 functions of yeast using a multilayer perceptron approach and reported that only 10% of these functions were trainable by learning algorithms [34]. Brown and Mateos used multiple microarray experimental data, by blindly combining or merging data sets based on same gene, the 6 data sets without

any selection of data sets in their learning procedures. **The first problem** we selected in our study is the problem of accurate functional classification of 5 functions of yeast genes using microarray data sets.

### 1.2.2 3 Types of Protein Sites

Several groups studied computational predictions of different protein binding sites, such as calcium binding sites [39, 31], serine protease active sites [58], ATP-binding sites [29], disulfide bond-forming sites sites [29] and other studies [55, 30, 68, 4]. For example, in [57], 19 features are used including physical, chemical and structural ones and classified into six categories: (a) chemical groups, (b) secondary structures,(c) atom-based, (d) residue-based, and (e) others. Wei et al. [31] built a classification model of calcium binding sites using the 19 micro-environment properties formulated by Steven et al. [57]. Wei et al. [31] used 16 calcium binding sites and 100 non-binding sites, and built a Bayesian classification model. The point to note is that Wei et al. [31] used 5 categories of micro-environment properties for prediction of calcium binding sites. The sixth category is Co-ordinates of the atom. One should note that the 6 sets of micro-environment properties are by nature very different from each other. They can be used either individually or in combination to infer if a candidate region in a protein is a protein binding site of specific type, i.e. calcium binding site. We pose a question whether it is necessary to use features from all these categories for the highly accurate predictions of protein binding site, or if it would be possible to achieve accurate predictions by using features coming from a restricted number of categories of micro-environment properties. Using as small as possible set of features for protein binding site predictions would be beneficial for the reason that not always all features, are available for the protein binding sites of interest. Moreover,

if the accurate predictions are possible using only information from restricted micro-environment properties categories, that would suggest importance of these categories of feature over others in the protein binding site recognition problems. **The second problem** we selected to study is the problem of building accurate classification models for 3 types of protein sites through micro-environment properties surrounding a site.

### 1.2.3 26 Specific Functions of Yeast Genes

To reveal the functional associations of genes in a genome, the gene expression profiles of a series of experimental assays or conditions can be analyzed to group the genes into clusters based on the similarity in their patterns of expression using machine learning techniques. These co-expression clusters can be interpreted as biological functional groupings for the genes—each cluster containing genes that encode proteins required for a common function. The functions of unknown gene products can then be systematically inferred through the *guilt-by-association* principle [64]. As genome-wide functional studies of genes become routine in biology laboratories, a rapidly increasing number of large gene expression data sets has now become accessible to researchers—either through collaborators or from online gene expression public repositories—for their biological investigations. We examine how to exploit this availability of microarray data resulting from multiple functional studies to build accurate functional classifiers for unknown genes. **The third problem** we selected to study is the problem of building accurate classification models on 26 specific functions of yeast genes with multiple microarray studies.

## 1.3 Result Summary

We undertook a research problem "Progressive Data Mining: An Exploration of Using Whole-Dataset Feature Selection in Building Classifiers on Three Biological Problems". Progressive Data Mining (PDM) demonstrates the usefulness of the combination of whole data sets selected by Hill or Greedy-Hill algorithms for building better classification models and the effects on accuracy. PDM, Hill, and Greedy-Hill are detailed in Chapter 5. We compare the results of this approach with other approaches—using the best of individual data sets, using all available data sets, using selected features from feature selection methods. We also evaluate how close our results are to the optimum result using the combination of whole data sets through exhaustive search. We focused on 3 bioinformatics problems—5 specific functions of yeast and 3 specific types of protein sites, and 26 specific functions of yeast genes.

### 1.3.1 Problem 1: 5 Functions of Yeast Genes

Spellman *et al.* [56] conducted microarray experiments on *S. cerevisiae* under different experimental conditions—$\alpha$-factor arrest, Cdc15 arrest, Elutriation, *Cln3* and *Clb2* activation—and monitored expression levels of 6221 genes at various time points. Similarly, Chu *et al.* [7] and DeRisi *et al.* [9] measured gene expression levels through sporulation and diauxic shift experiments respectively. Thus there are 6 sets of gene expression experiments, one for each of the 6 experimental conditions mentioned here. Eisen *et al.* [10] combined these 6 sets of gene expression experiments, and performed a clustering of 2467 yeast genes based on their gene expression values in these experiments. They showed that genes that share a common cellular function would exhibit similar gene expression profiles. Building on the work of Eisen *et al.*, and

with some modifications to Eisen's data, Brown *et al.* [5] attempted to make inference on 5 specific cellular functions of yeast genes based on gene expression profiles. The 5 specific cellular functions considered by Brown *et al.* are those pertaining to TCA cycle, respiration, ribosomes, proteasomes, and histones. Brown *et al.* also proposed the performance measure $S(M)$ to evaluate a classification model $M$.

We show that we can much more accurately infer whether a gene is involved in the 5 specific cellular functions, if we use these 6 data sets in combination opposed to using any single one of them. Our results show that using multiple data sets in combination has 26% chance of yielding better results than using the best of individual data sets. We also show that we can infer more accurately whether a gene is involved in the 5 specific cellular functions, when we use some combination of data sets but not necessarily all the available data sets. Our results show that using multiple data sets in combination has 26% chance of achieving better results than using all available data sets. We also show that feature selection methods can yield better results than using the best individual data sets or using all available data sets. We also show for 60% (60%, 80%, and 80%) of the protein functional classes, we are able to use a combination of 2 or more whole data sets to obtain a higher prediction accuracy than using the best performance from feature selection methods, through C4.5 (SVM, NBay, and MLP, respectively). Even though using conventional feature selection approach gives a significant improvement compared to using the best of individual data sets and using all data sets blindly, it does not lead to the best accuracy often enough for the 5 functions. Our results show that the combination of whole data sets chosen by Hill (we will describe this method in Chapter 5) achieves better results. Hill is better in 60% (60%, 40%, and 60%) of protein functional classes than the

best individual data method through C4.5 (SVM, NBay, and MLP, respectively). Similarly, Hill is better in 80% (80%, 60%, 60%) of the cases than by the all data method and in 40% (60%, 80%, and 40%) of the cases than the feature selection method by same algorithms. We also show results from Greedy-Hill (we will describe this method in Chapter 5). When results of feature selection methods are compared with that of Greedy-Hill, we found that Greedy-Hill achieves better results in 11 out of 20 cases, equal results in 4 out of 20 cases, and lesser results in 5 out of 20 cases. This means Greedy-Hill is capable of achieving a better or equal performance by $S(M, 2)$ in 15 out of 20 cases. Finally we show that the combinations chosen by Hill are among the 7% of combinations (and by Greedy-Hill, among the 8% of combinations) that give the best performance, at least for the purpose of predicting the 5 specific functions of yeast genes. We show that Greedy-Hill is much faster in selecting important data subsets than exhaustive search and Hill. In fact, a typical run of Greedy-Hill would take 1367 seconds, compared to 1726 seconds for Hill and 55308 seconds for exhaustive search. The average performance on 5 functions of yeast by Greedy-Hill is an S(M) score of 52.60, Hill is 53.80, and Exhaustive search is 56.55. Thus Hill should be used for classification problems where a small number of data subsets are considered, but Greedy-Hill should be used where a larger number of data subsets are encountered.

### 1.3.2 Problem 2: 3 Types of Protein Sites

Bagley *et al.* [57] characterized and formulated micro-environment features surrounding protein sites. These features are based on inherent properties that can be calculated from the atoms defining a protein site and its neighborhood. Wei *et al.* [31] studied calcium binding site, Bagley *et al.* [58] studied serine protease active site, and

Wei *et al.* [29] studied ATP-binding site and disulfide bond-forming sites.

We show that we can much more accurately infer whether a candidate region is a calcium binding site, a serine protease active site, or a disulfide bridge, using multiple sets of micro-environment properties than using any single set of micro-environment properties. We show that we can much more accurately (31% chance of yielding better results) infer if a candidate region is a calcium binding site, a serine protease active site, or a disulfide bridge, using a combination of 2 or more of micro-environment properties—but not all available data sets—than using all available data sets of micro-environment properties. Our results show that 10% of the possible combinations of sets of micro-environment properties yield better accuracy than using all data. We show that feature selection methods can yield better results than using the best of the individual sets of micro-environment properties or using all sets of micro-environment properties. We also show for 100% (100%, 100%, and 67%) of the types of protein sites, we are able to use a combination of 2 or more sets of micro-environment properties to obtain a higher prediction accuracy than the best performance from feature selection methods, through C4.5 (SVM, NBay, and MLP, respectively). Thus, while the conventional feature selection approach is a significant improvement over the use of the best of individual data sets and over the use of all data sets blindly, it does not lead to the best accuracy often enough for this protein site classification problem.

Our results show that combination of whole sets of micro-environment properties chosen by Hill (we will describe this method in Chapter 5) achieve better results. Hill is better in 100% (100%, 67%, and 100%) of types of protein sites than best individual micro-environment method, through C4.5 (SVM, NBay, and MLP, respectively).

Similarly Hill is better in 67% (100%, 67%, 67%) than the All sets data method and in 100% (67%, 100%, and 33%) than the feature selection data method by same algorithms. We also show results from Greedy-Hill (we will describe this method in Chapter 5). When results of feature selection methods are compared with that of Greedy-Hill, we found that Greedy-Hill achieves better results in 6 out of 12 cases, equal results in 2 out of 12 cases, and lesser results in 4 out of 12 cases. This means Greedy-Hill is capable of achieving a better performance in 8 out of 12 cases. Finally we show that the combinations chosen by Hill are within the 2% of combinations (and by Greedy-Hill, among the 8% of combinations) that give the best performance at least for the purpose of predicting the 3 specific types of protein sites. We show that Hill is much faster in selecting important data subsets than exhaustive search and Greedy-Hill for this problem. In fact, a typical run of Greedy-Hill would take 94 seconds, compared to 90 seconds for Hill and 475 seconds for exhaustive search. The average performance on 3 types of protein sites by Greedy-Hill is a S(M) score of 101.00, Hill is 105.83, and Exhaustive search is 106.67. Thus Hill should be used for classification problems where a small number of data subsets are considered, but Greedy-Hill should be used where a larger number of data subsets are encountered.

### 1.3.3 Problem 3: 26 Functions of Yeast Genes

SMD (Stanford Microarray Database [16]) contains a huge collection of microarray gene expression data sets based on several experimental conditions. We retrieved 16 data sets from SMD. 6 of the 16 data sets contain multiple wet experiments conducted under different experimental conditions. So, we partition these 6 data sets into 47 data sets based on individual experimental conditions. Now, we have a total of 57 data sets. The next step is to consider functions for our classification study. The

MIPS catalogue dated 19 March 2004 (Version 2.0) has 116 functions at the second level functional annotations of genes. After removing functions which contain less than 25 genes, 26 functions are left. We report only results on 26 functions of yeast in this thesis.

Exhaustive search is not feasible over 57 data sets, as such an exhaustive search is computationally expensive ($2^{57} - 1$ possible combinations to consider). We use the combinations of whole data sets from Greedy-Hill (we will describe this method in Chapter 5) method and compare the results with that of using the best of individual data sets, using all available data sets, and using selected features from feature selection methods. We show that for many of the 26 functional classes, we can find a combination of data sets from the 57 different experimental conditions that yield better accuracy than using the best of all single data sets. Results show that for 30% (33%, 26%, and 43%, respectively) of the protein functional classes, the use of additional data sets (on same set of genes) lead to a better prediction accuracy than using the best of individual data sets through C4.5 (SVM, NBay and MLP, respectively). We show that for most of the 26 functional classes, we can find a combination of data sets from the 16 different experimental conditions that yield better accuracy than using all the 16 data sets together. Results show that for 63% (83%, 93%, and 76%, respectively) of the protein functional classes, the use of a careful combination of data sets leads to a better prediction accuracy than using all available data sets through C4.5 (SVM, NBay and MLP, respectively). We show that feature selection methods can yield better results than using the best individual data sets or using all available data sets. We also show for at least 37% (43%, 72%, and 61%, respectively) of the protein functional classes, we are able to use a combination of 2 or more data sets,

to obtain a higher prediction accuracy than using the best performance from feature selection methods through C4.5 (SVM, NBay, and MLP, respectively). So, while the feature selection approach is a significant improvement over the use of the best of individual data sets and over the use of all data sets blindly, it does not lead to the best accuracy often enough for this protein function problem. The average performance on 26 functions by Greedy-Hill is a S(M) score of 15.1, Hill is 13, all data sets (ALL) is −32.12, best of individual data set (BI) is 9.8, and selected features from correlation feature selection (CFS) is 9.4. Thus Hill should be used for classification problems where a small number of data subsets are considered, but Greedy-Hill should be used where a larger number of data subsets are encountered.

**Keywords :** Progressive Data Mining, Microarray, Functional studies, Multiple datasets, Feature selection, Support Vector machines, Multilayer perceptron, Multi-class classification, Correlation-based feature selection, Chi-square, Information-gain, Whole Dataset Feature selection, Binding sites, Hill climbing algorithm, Greedy-Hill climbing algorithm, Neural network, C4.5, Naïve bayesian.

**Organisation on Thesis Report :**

Chapter 2 is a brief survey on functional classification problems through existing methods.

Chapter 3 describes data sets for the 3 research problems taken in our study. We briefly explain the differences on the yeast Catalogue (19-March-2004, Version 2.0 used in our study) and the new yeast Catalogue, Version 2.1 dated 9th January,

2007.

Chapter 4 explores existing methods—using the best of individual data sets, using all available data sets, using selected features from conventional feature selection methods, using exhaustive search.

Chapter 5, illustrates the concept of "Progressive Data Mining" through "Whole Dataset Feature Selection Algorithms"—"Hill climbing method" (*Hill*) and "Greedy-Hill climbing method" (*Greedy-Hill*). In Chapter 6, we discuss effects of using "Combination of features" and applying "Committee methods" in building classification models. We further list follow-up research work to consolidate the focus of our research, as well as future directions enabled by this thesis.

In Appendix A, B, and C, tables for 5 functions of yeast genes, 3 types of protein sites, and 26 functions of yeast genes are listed, respectively.

We could not tabulate some results—additional tables for 5 functions of yeast genes, 3 types of protein sites, and multiple evaluation metrics tables for 26 functions of yeast genes, tables for 20 other functions of yeast genes—due to space constraint.

# Chapter 2

# Survey of Existing Methods

We show different classification problems that are considered in this research study. We briefly summarise previous studies on these specific problems to help understand subsequent chapters of this report, where we will discuss our methods and results.

Specifically:

- Eisen *et al.* [10] studied wet lab analysis of 5 cellular functions of *S. cerevisiae* genes under 6 experimental conditions. We discuss functions and different experimental conditions in which gene expression assays are recorded. These 5 functions are used in classification studies [5] through learning algorithms.

- Wei *et al.*, [31] studied calcium binding sites through attributes generated by FEATURE package with scoring function based on Bayesian. FEATURE package computes a score for a given query region. Score value tells whether the query region is a calcium binding site or not. They [31] used 16 calcium binding sites and 100 non-binding sites in their model through a 3-fold cross validation scheme by Bayesian.

- Mateos *et al.* [34] conducted functional study on 96 functions of yeast genes

with the 6 sets of microarray data of Eisen *et al.* [10] and reported that only 10% of functions are trainable through learning algorithms.

## 2.1 The Study on Functions of Yeast Genes

Our living cell is a complicated system comprising multiple cellular pathways performing different biological functions dynamically. Informatics research in the biomedical and life sciences domain has revolved around large growing databases of scientific literatures, DNA sequences and protein structures. Entries in the international repository of biological sequences, GeneBank, now surpass 30 million, although it took 18 years for the database to reach its first 10 million entries in 2000. This exponential growth of biological sequences in recent years has been partially fueled by the completion of many genomic sequencing projects which have identified many novel genes with unknown functions.

Elucidating the biological roles of these novel genes has become the main challenge in the post-genomic era. Many researchers have exploited the availability of context information in complete genomes, from which the novel genes are derived, for assigning putative function to novel genes. Examples of these context-based methods include gene fusion, gene locality and phylogenetic profiling. These methods depend on the expression of the specific biological phenomena which make them applicable only for a subset of the novel genes. Microarray data, on the other hand, is another growing biological data type that offers richer information than genomic sequences and can theoretically be used to assign putative function of all novel genes in a genome. Through genome-wide measurements of mRNA expression levels across multiple experimental conditions, we can obtain global snapshots of the cell's genetic

activities at various stages and in different conditions. We can then use these gene expression data to elucidate the functional roles of the various genes as they partake in the underlying biological pathways.

One common approach in functional analysis of gene expression data is *clustering*—organizing genes into different functional groups based on the principle that genes belonging to the same functional groups or pathways will have similar expression profiles over a range of experimental conditions. One major drawback of clustering approaches is that the groupings are learned directly from the expression data [6, 10] without taking advantage of the often available predefined class information. As a result, clustering approaches can generate clusters of genes that do not correspond well to the true underlying biological pathways.

Biologists often have previous knowledge that a subset of genes is involved in a biological pathway. They have interest in discovering other genes which can be assigned to the same pathway. *Classification* approach is more suitable than clustering for functional classification of genes using microarray data. Unlike clustering, classification can build a model with known biological knowledge and classify new genes based on the model. Supervised classification learning algorithms tend to assign pathway memberships that correspond well to the true underlying biological pathways.

### 2.1.1 Microarray Experiments

Large scale analytical methods such as cDNA microarrays for global gene expression profiling and tissue microarrays for simultaneous analysis of individual markers in multiple samples have improved our ability to understand biological defects that occur in cancer development. Although many of the microarray experiments may have been conducted on identical sets of genes, the studies are often designed to

address different scientific investigations, usually conducted under different experimental conditions. For example, one microarray experiment is focused on identifying new components in polyphosphate metabolism using the gene knockout method—the PHO regulatory pathway is involved in the acquisition of phosphate ($P_i$) in *S. cerevisiae*. When extra cellular $P_i$ concentrations are low, several genes are transcriptionally induced by this pathway which includes the $Pho4$ transcriptional activator, the $Pho80 - Pho85$ cyclin-CDK pair, and the $Pho81CDK$ inhibitor. In an attempt to identify all the components regulated by this system, a whole-genome DNA microarray analysis was employed, and 22 PHO-regulated genes were identified [40]. Similar microarray experiment on the same set of genes may be designed to study spore morphogenesis by times series investigation such as [7] as explained below in experimental conditions and objectives. Fernandes et.al [11] studied yeast genes with "high hydrostatic pressure" experimental condition, and identified 274 genes belonging to "stress response" function. Biological conditions are altered or aimed to achieve their specific objective of finding a set of genes in specified functional pathways. Intuitively, it should be beneficial to combine the two-expression datasets, given that they have been conducted on the same set of genes (both cited experiments used the *Saccharomyces cerevisiae*'s genome in their investigations). On the other hand, the differences in their study objectives and experimental conditions (explained in experimental conditions and objectives) may not warrant that combining or merging data from these different studies can give rise to better or new information.

Microarray is a growing biological data type that offers richer information than genomic sequences. Theoretically it can be used to assign putative function of all novel genes in a genome through machine learning methods. Through genome-wide

measurements of mRNA expression levels across multiple experimental conditions, global snapshots of cell's genetic activities at various stages are obtained. Gene expression data is used to elucidate functional roles of genes as they partake in the underlying biological pathways.

Function is attached to the gene expression data for each gene. For example genes involved in "respiration activities" get their group as "respiration". Genes for whom functional annotations are not experimentally available are treated as "unknown" group. Machine learning algorithm is used on known and unknown samples of data to build functional classification models. Previous researchers [21, 60, 24] focused on mining microarray data with individual experimental data only.

Selecting appropriate datasets for functional analysis is becoming more crucial as some microarray data is of poor quality. Multiple microarray datasets on the same set of genes can often be collected from different laboratories and research centers, either through collaborators or from online gene expression data repositories. It is obviously useful if we can effectively combine these additional data sets (on same set of genes) with the data generated in our own laboratory to further improve our functional study of genes, and to make deductions that we cannot make using our own data alone. But inclusion of noisy data with different experimental objectives may create interference to functional analysis. For example, finding genes involved in the anaerobic metabolism may not be possible unless under experimental condition where the cells are deprived of their normal energy source. Researches focused on mining microarray data with individual experimental data only [21, 60, 24]. Brown *et al.* [5] and [34] used combined microarray data set in their learning procedures. Mateos *et al.* [34] reported only 10% of yeast functions are trainable through learning algorithm due to

noisy nature of gene expression data set.

We note that previous work considered either individual data set or blindly combined all data sets approach in their classification studies. In this study, we show use of partitioning data set based on experimental conditions and selection of whole data set based on their usefulness in classification model. We considered many functions of *Saccharomyces cerevisiae* from Munich information for protein sequences and gene expression data sets from Stanford Microarray Database with different experimental conditions.

### 2.1.2 Application of Machine Learning Approaches

Machine learning algorithms are regularly applied in genome laboratories for building functional classification models on genes. In this section we describe recent studies on functional classification of *Saccharomyces cerevisiae* with gene expression by learning algorithms.

Spellman [56] measured relative levels of mRNA as a function of time in cell cultures, synchronized in three independent ways by using $\alpha$ pheromone, centrifugal elutriation and cdc-15-2 to arrests cell growth. Cln3 and Clb2 with glucose addition is used to arrest cell growth. Gametogenesis in yeast involves two overlapping processes, meiosis and spormorphogenesis, involved in differential gene expression. An experimental condition is set by Chu [7] using nitrogen-deficiency growth media to induce sporulation of yeast. A recurring cycle in the natural history of yeast involves a shift from anaerobic (fermentation) to aerobic(respiration) metabolism. Switch from anaerobic growth to aerobic respiration upon depletion of glucose, referred to as the diauxic shift, is correlated with widespread changes in the expression of genes involved in fundamental cellular processes such as carbon metabolism, protein synthesis and

carbohydrate storate [9].

Eisen *et al.* [10] studied how cellular functions of *S. cerevisiae* are affected under 6 experimental conditions. This resulted in 6 individual gene expression data sets—$\alpha$-factor, Cdc15, Elutriation, *Cln3* and *Clb2*, Sporulation, Diauxic shift, on 6221 genes of *S. cerevisiae*.

Spellman *et al.* [56] suggested that some data points could be removed as they were aberrant. Brown *et al.*, removed 15 time scale points {Cdc15:10, Cln3-Clb2:3, Sporulation:2} from Eisen *et al.*, data set on yeast and added 14 time scale points (HEAT:6, DTT:4, COLD:4). Brown *et al.*, used MIPS functional annotation dated 08th Nov, 2001 and applied learning algorithms on 5 functions—{Tri-Carboxyilic Acid,:17, Respiration:30, Cytoplasmic ribosomes:121, Proteasome:35, Histones:11}.

Brown *et al.* [5] reported that support vector machine algorithm with higher degree kernel function gives better performance than other algorithms on 5 functions of yeast genes. Brown *et al.* [5] developed GIST (http://microarray.cpmc.columbia.edu/gist), Version 2.0.5. Support vector machine algorithm is tuned with polynomial kernel of degree 3 which gives better classification performance.

Vert *et al.* [62] related gene expression profiles to signalling pathways thereby encoding both gene network and expression profiles into two kernel functions and perform regularised form of correlation analysis between two kernels. Method applied on Alpha factor data set (Spellman *et al.*, 1998) and diauxic shift data set (DeRisi *et al.*, 1997) provides a way to compare a graph of metabolic pathways to a set of expression profiles.

Previous studies show difficulties in building functional classification with gene expression profiles sets. Genes with related functions tend to be expressed in similar

patterns. This can suggest possible roles of genes of unknown functions based on their temporal association with genes of known functions. The **objective** is to measure expression levels of yeast genes at various time points based on individual experiment. In our work, we consider the entire dataset from each study to be one feature. We then devise a whole-dataset feature selection method to decide on the appropriate microarray datasets to be combined for improved functional analysis. We use a simple Hill climbing method for whole-dataset feature selection, and show that it can better improve data analysis results from multiple microarray datasets.

## 2.2 The Study on Protein Sites

Three dimensional structures of proteins are determined by X-ray crystallography, nuclear magnetic resonance (NMR), and homology modeling through computational tools. It is an important step to recognize functional roles and conserved sites like—active sites, binding sites, and structural support sites present in proteins. It is important to determine binding sites of a protein structure more accurately in order to efficiently design drugs. Analyzing known structures and finding protein sites manually is not feasible. Thus computational tools are needed. Recently, many researchers have attempted recognition of protein sites by using micro-environment properties surrounding candidate sites through learning algorithms.

### 2.2.1 Micro-environment Properties

Once complete structure information of a protein is known, a protein site can be defined by the three dimensional location co-ordinates of its key atoms and the neighborhoods around these locations. Bagley *et al.* [57] characterized and formulated

micro-environment features surrounding protein sites. These features are based on inherent properties that can be calculated from the atoms defining a protein site and its neighborhood. This resulted in 6 categories of micro-environment properties comprising—Atom (4 atomic types and charges), Chemical (6 chemical-based properties), Residue (3 residue type classifications), Sec-str (2 secondary structure classifications), Others (4 general properties and measures), and Co-ord (3 3-dimensional co-ordinates of atoms). These 22 micro-environment properties are detailed in Table 3.3. These properties can be used to study protein sites—CALCIUM (calcium binding sites), DISULFIDE (disulfide bridges) and SERINE (serine protease active sites)—through learning algorithms.

Yamashita *et al.* (1990) studied protein sites for calcium binding which are centered in a shell of hydrophilic residues. Based on this research Nayal and Di cera (1994) proposed a valency function that can predict calcium binding sites with spatial accuracy. Recently, due to heavy experimental study on three-dimensional protein structures we have an opportunity to model statistical methods for characterizing and recognizing calcium binding sites[31]. Bagley and Altman (1995) proposed a system—FEATURE—which extracts many properties from three-dimensional geometry, residue and other structural information. These features are based on physical, chemical and atomic information from protein data bank.

Bagley *et al.* [57] showed that the distributions of micro-environment properties differ significantly between sites and non-sites. They reported that CALCIUM possessed statistically significant excess of negatively charged, acidic, oxygen-rich, mostly Asp and Glu moieties at radii 2–7$\dot{A}$. In DISULFIDE, local hydrophobicity was low in shells at 0–3$\dot{A}$ surrounding disulfide bonds; B-factor for the neighborhood was lower

than controls; and increased polar effect atoms at 1–3$\dot{A}$. In SERINE, a significant range of shells (0–7$\dot{A}$) contained atoms forming 3-helices; and showed solvent accessibility in the immediate neighborhood of the His (shells 0-3) compared to non-sites.

Wei *et al.* find calcium binding sites through the attributes generated by FEATURE package[31]. The goal for them is to compute a score that will tell whether a query region is a calcium-binding site or not. They use a scoring function based on Bayesian. Wei *et al.* use 16 calcium binding sites and 100 non-binding sites in their study to model calcium binding sites by Bayesian algorithm and reported that sensitivity and specificity are as high as 90%. Wei *et al.* [31] took micro-environments properties formulated by Bagley *et al.* [57] as features in machine learning algorithms to study calcium vs non-sites. Wei *et al.* ignored the 3 three-dimensional co-ordinates features and used remaining 19 features from Bagley *et al.* to build their Bayesian classification model.

Bagley *et al.* [58] used micro-environment properties to study 6 molecules for serine protease active sites, as an extension to their earlier study [57]. In this study, they suggested a general purpose method of modular property representation that could be used to analyse any kind of macromolecule. They reported that property distributions are within a reasonable statistical framework.

Wei *et al.* [29] improved a Bayesian system called FEATURE to characterize and recognize geometrically complex and asymmetric sites such as ATP-binding site and disulfide bond-forming sites. Adenosine triphosphate (ATP) molecule plays vital roles in energy transfers in living systems. Wei *et al.* created statistical profiles that distinguished ATP-binding sites from random non-sites in 3D structures and later used these profiles to recognize new ATP-binding sites. Wei *et al.* analysed

disulfide bond-forming sites and redoxin active site. In a protein structure, a cysteine residue can appear either in free form or covalently bonded to another cysteine via a disulfide group. A redoxin active site—including thioredoxins and glutaredoxins—are small proteins that participate in thiol-disulfide exchange reactions via the reversible oxidation of an active central disulfide bond. 90 disulfide-bonding cysteines from 16 non-redundant proteins and 48 free cysteins—as non-sites—from 19 non-redundant proteins were used in their study.

## 2.3 The Study on Functions of Yeast Genome

Experimental biology findings has its limitation. Even for simple organism like yeast which can grow fast and can be subjected for repeated study due to low cost and time, has only about 50% of genes with confirmed functional annotations. This limited knowledge warrants computer scientists to model biological data using machine learning approaches. It subsequently helps biologists to have a better goal in their search of functional pathway of all genes. Building accurate classification models with limited knowledge is a challenging problem as many learning algorithms prefer more samples than experiments. On the other hand, we usually have less samples than experimental assays. If we blindly combine more data sets, it will be over fitting the data and all learning algorithms cannot build a proper classification model.

### 2.3.1 Multiple Microarray Data Sets

Spellman *et al.* [56] conducted microarray experiments on *S. cerevisiae* with experimental conditions—$\alpha$-factor arrest, Cdc15 arrest, Elutriation, *Cln3* and *Clb2* activation—and monitored expression levels of 6221 genes at various time points.

Similarly, Chu *et al.* [7] and DeRisi *et al.* [9] measured gene expression levels through sporulation and diauxic shift experiments respectively. Thus there are 6 sets of gene expression experiments, one for each of the 6 experimental conditions mentioned here. These 6 sets of gene expression experiments comprise 80 individual gene expression experiments, whose details will be provided in Table 3.1.

Mateos et al [34] use Spellman *et al.* [56] data sets and 96 functions of yeast (from MIPS catalogue dated November 8, 2001). They used MLP (multilayer perceptron) in their classification study. Performance was measured as true positive rate (TP = Number of true positive genes from a classifier / Number of actual positive genes in data set), in a three-fold cross validation scheme. They reported more than 60% false negative on 92% of the functions studied. They further reported that correlation between class size and learning rate is less than perfect—for example, "glyoxylate" achieved high TP = 35% with only 5 genes while "biogenesis of cell wall" achieved TP = 4% with 85 genes. They also reported that a very faint trend for smaller classes to be more heterogeneous, and larger classes to be more homogeneous. They applied an iterative procedure on the function TCA and reported that only with a threshold $\tau = 0.8$, their classifier achieved optimum performance. Finally, Mateos *et al.* concluded that only 10% of functions are trainable through learning algorithms with gene expression data sets.

Clare *et al.* [8] use the "cellcycle" data set from Spellman *et al.* (1998), "church" data set from Roth *et al.* (1998), "derisi" data set from DeRisi *et al.* (1997), "eisen" data set from Eisen *et al.* (1998), "gasch1" data set from Gasch *et al.* (2000), "gasch2" data set from Gasch *et al.* (2001), "spo" data set from Chu *et al.* (1998), and "expr" is formed by combining or merging these 7 microarray data sets. As our

research is focused on functional classification of yeast genes through microarray data sets only, we discuss results related to microarray data sets only. They used DMP (Data Mining Prediction), and two complementary forms of data mining—Inductive Logic Programming (Muggleton *et al.*, 1992) and propositional rule learning (Mitchell, 1997). Average accuracy of 33% on the "cellcycle" data set, 51% on "derisi" data set, 40% on "eisen" data set, 63% on "spo" data set, and 37% on "expr" data set, are achieved for second level functional annotations based on the MIPS catalogue dated April 24, 2002 on their test data. Please refer to their paper [8] for other data sets and results.

Genome laboratories regularly conduct microarray gene expression experiments on yeast. This results in having many gene expression data sets from each experiment. These data sets are regularly deposited into SMD (Stanford Microarray Database [16]). In this way, SMD possesses a huge collection of microarray gene expression data sets based on several experimental conditions. Researchers can download these data sets for building functional classification models.

# Chapter 3

# Description of Data Sets and Methods

In Chapter 2 the research studies of Brown [5], Wei [31], and Mateos [34] are summarised. In this chapter we give details of our data sets on the three problems—the study of 5 functions of yeast genes through 6 gene expression data sets, the study of 3 types of protein sites through 6 types of micro-environment properties, and the study of 26 functions of yeast genes through 57 data sets.

Specifically:

- In the previous chapter we detailed the work of Eisen *et al.* [10] who studied wet lab analysis of 5 cellular functions of *S. cerevisiae* under 6 experimental conditions. These 5 functions are used in classification studies [5] through learning algorithms. In this chapter we give details of data sets used in our research study.

- In the previous chapter we described the study on calcium binding sites through attributes generated by FEATURE package with scoring function based on Bayesian by Wei *et al.*, [31]. In this chapter we give details on how we derived

the data sets for our research study on modeling 3 types of protein sites.

- In the previous chapter we described the study by Mateos *et al.* [34] on 96 functions of yeast genes with the similar 6 sets of microarray of Eisen *et al.* [10] and reported that only 10% of functions are trainable through learning algorithms. In this chapter we give details of the 16 data sets used in our research study on modeling 26 functions of yeast.

## 3.1 Yeast Genes

Eisen *et al.* [10] showed that genes involved in common functional activities would have similar gene expression profiles under a number of experimental conditions. Brown *et al.* [5] studied—in a follow-up to Eisen *et al.*—how 5 specific cellular functions of *S. cerevisiae* are affected under 6 experimental conditions. This resulted in 6 gene expression data sets described in Table 3.1. These data sets can be used, either individually or in combination to infer if a gene is involved in the 5 specific cellular functions studied.

### 3.1.1 6 Gene Expression Data Sets

We use the gene expression data of *Saccharomyces cerevisiae* from six different microarray studies available from (http://rana.lbl.gov/EisenData.htm) Eisen's Lab [10]. These six microarray studies have been performed on the same set of genes from yeast, but with different experimental objectives and under varying experimental conditions. Table 3.1 shows the major differences between the six datasets *Alp*[56], *Cdc* [56], *Elu* [56], *Ccc* [56], *Spo* [7], and *Dia* [9]. Collectively, the six datasets comprise gene expression vectors from a total of 80 experiments on 6,221 yeast ORFs. Out of the

6,221 genes used in the experiments, 2,550 are known yeast genes with annotated functions in MIPS (Munich Information Center for Protein Sequences) [36].

Table 3.1: 6 microarray data sets used in our study.

| Study | Experimental condition | Experimental objective | Time Points |
|---|---|---|---|
| *Alp* | $\alpha$ factor-based synchronization | cell cycle | 18 |
| *Cdc* | *Cdc15*-based synchronization | cell cycle | 25 |
| *Elu* | elutriation synchronization | cell cycle | 14 |
| *Ccc* | *Cln3* and *Clb2* experiments | cell cycle | 3 |
| *Spo* | nitrogen deficiency | spore morphogenesis | 13 |
| *Dia* | glucose depletion | diauxic shift | 7 |

### 3.1.2 5 Specific Functional Annotations of Yeast Genes

For a more comprehensive study, we also apply our method to all the MIPS-annotated yeast genes in non-singleton functional classes (i.e., functional classes with more than one genes). Unlike previous similar studies such as the study by Mateos *et al.*, we chose to exclude genes with ambiguous functional assignments—namely, genes that belong to multiple functional classes—as we observe that the inclusion of such genes in the training process can affect the results, causing deterioration of the classifiers learned (data not shown). Out of the 2,550 annotated yeast genes in our expression datasets, there are 1,851 genes unambiguously assigned to a total of 60 non-singleton MIPS functional classes and available for our comprehensive evaluation study. Classification problem needs a label to be assigned to a set of samples for recognizing them. Functional annotations of yeast genes (TCA, ribosomal, histone, respiration and proteasome) are attached as class labels to microarray datasets as detailed in Table 3.2 for solving functional classification problems. For the known functional

classification of the 2,550 annotated genes, we refer to functional assignments provided with Eisen's data [10] which is based on the *Comprehensive Yeast Genome Database* (CYGD catalogue version 1.3, dated 25th June, 2003 [36]) from MIPS. The CYGD is a yeast gene annotation database based on extensive knowledge extracted from literature.

Table 3.2: 219 yeast genes on 5 functional classes from MIPS.

| Function | Description | Number of genes |
|---|---|---|
| TCA | Tricarboxylic acid cycle | 22 |
| RESP | genes in respiratory processes | 24 |
| RIBO | ribosomal genes | 129 |
| PROT | genes of the proteasome | 33 |
| HIST | histone-related genes | 11 |

For comparison, we focus on the five different MIPS classes that both Brown *et al.* [5] and Mateos *et al.* [34] had analyzed previously. The five classes are shown in Table 3.2 while many functional classes can be unlearnable [34], these five functional classes are proved to be machine-learnable by several previous studies [10, 5, 34]. Biologically, they represent categories of genes expected to exhibit similar expression profiles on biological grounds. While data from different experiments keep accumulating, it is essential to have accurate means for extracting biological significances and using the data to assign functions to genes. Microarrays continue to have limitations in addition to their technical difficulty, specificity and reliability: Microarrays are victims of their own success since the large data sets generated by these chips add new statistical and informatics-related challenges and complexity. The process of extracting accurate knowledge becomes more challenging for classification algorithms.

## 3.2 Types of Protein Sites

Protein sites are micro-environments within a biomolecular structure, distinguished by their structural or functional role. Steven *et al.* [57] formulated 6 sets of micro-environment properties—comprising co-ordinates, chemical groups, secondary structures, atom-based, residue-based, and other properties—surrounding a protein site. They showed that the distributions of these properties differ significantly between sites and non-sites. In particular, Steven *et al.* studied calcium binding sites, serine protease active sites, and disulfide bridges. Subsequently, Wei *et al.* [31] built a classification model on calcium binding site using the 19 micro-environment properties formulated by Steven *et al.* as features. Wei *et al.* took 16 calcium binding sites and 100 non-binding sites, and built a Bayesian classification model. The point to note is that Wei *et al.* used 5 categories of micro-environment properties in inferring calcium binding sites. Again, these 6 sets of micro-environment properties are very different from each other in nature. They can be used either individually or in combination to infer if a candidate region in a protein is a calcium binding site.

### 3.2.1 6 Micro-Environment Properties

Protein sites are micro-environment within a segment of biomolecular structure of a protein, recognized by their functional role. Bagley *et al.* characterized and formulated micro-environment features surrounding protein sites [57]. This resulted in 6 categories of 22 micro-environment properties comprising—Atom (4 atomic types and charges), Chemical (6 chemical-based properties), Residue (3 residue type classifications), Sec-str (2 secondary structure classifications), Others (4 general properties

and measures), and Co-ord (3 3-dimensional co-ordinates of atoms). These 22 micro-environment properties are detailed in Table 3.3.

Table 3.3: 6 categories of micro-environment properties.

| Category | Properties in each category |
|---|---|
| Co-ord | X-Co-ord, Y-Co-ord, Z-Co-ord |
| Atom | Atom type, Hydrophobicity, charge, charge-with-His |
| Chemical | Hydroxy, amide, amine, carbonyl, ring-system, peptide |
| Residue | Type, class1, class2 |
| Sec-str | Secondary structure, strclass1, strclass2 |
| Others | VDW-volume, B-factor, mobility, solvent accessibility |

### 3.2.2 3 Types of Protein Sites

In this study, 3-dimension structural data are retrieved for proteins from Protein Data Bank. Table 3.4 gives 19 protein IDs on 3 protein sites—CALCIUM (calcium binding site), SERINE (serine protease active site), and DISULFIDE (Disulfide bridge). These proteins are biologically known to consist the 3 protein sites.

Table 3.4: Proteins on 3 types of protein sites from PDB.

| Protein sites | Sample size | PDB ID |
|---|---|---|
| `CALCIUM` | 94 | 1NPC 1TMN 2MSB 3LHM |
| `SERINE` | 43 | 1GCT 1SGT 1TON |
| `DISULFIDE` | 37 | 2IG2 2PRK 2SN3 3GRS 6PAD |

**Steps in building data sets :** 3-dimensional protein structural data for the proteins (Table 3.4) are retrieved from Protein Data Bank. "CASTp" (Computed Atlas of Surface Topography of proteins [3]) identifies all pockets and cavities for a three-dimensional structure of a protein. It also measures their volume

and area analytically and give the number, area, and circumference of the mouth openings for each pocket. Pocket information files are retrieved for a type of proteins sites—$s \in$ {CALCIUM, SERINE, DISULFIDE}—through the "Castp" server (http://sts.bioengr.uic.edu/castp/index.php) [25, 26, 27]. From pocket information files known types of protein sites information are extracted. Three Co-ordinate values are taken from 3-dimensional co-ordinates of PDB. The remaining 19 micro-environments properties are calculated from atom records (as formulated by Wei [29]).

## 3.3 Yeast Genome

Mateos *et al.* [34] used the gene expression data sets of Eisen *et al.* [10] and tried to predict 96 functions of *S. cerevisiae* using multilayer perceptron. They reported that only 10% of functions are trainable by their approach—this is not surprising since many of the 96 functional classes have too few members or have ambiguous members.

### 3.3.1 57 Multiple Gene Expression Data Sets

The Stanford Microarray Database (SMD) possesses a huge collection of microarray data from global laboratories. In this subsection we detail on 16 microarray datasets that are retrieved with different experimental conditions for 6443 genes, from SMD database (http://genome-www5.stanford.edu/). Missing values for some experiments are left as is, because our data mining software is capable of handling them effectively.

6 of the 16 data sets each contains wet experiments where the experimental conditions are different. Original authors grouped different datasets as one single dataset. Our focus of research is to validate the use of individual data set based on different experiment. So, we partition these 6 data sets into 47 data sets based on individual

Table 3.5: 16 microarray data sets from SMD.

| Code | Experimental condition | Experimental objective | Data Points |
|---|---|---|---|
| *Alp* | Alpha factor arrest, Cln3-Clb2 activation, Elutriation, Cdc15 arrest | Identify genes in cell cycle pathways [56] | 59 |
| *Ace* | Transcriptional activators Ace1 and Mac1 | Identify genes expressed under growth conditions of excess copper or copper deficiency [18] | 6 |
| *Des* | DES460 and 0.02%mms | Role of the Mec1 pathway in modulating the cellular response to DNA damage [14] | 41 |
| *Haa* | Haa1 regulated transcription | Finding genes encoding membrane proteins [22] | 4 |
| *Hea* | Heat shock | Finding genes with similar drastic response to many environmental changes [13] | 133 |
| *Dby* | DBY8778 stationary phase | Identifying regulatory modules from gene expressions [52] | 33 |
| *Cal* | Calcium time series | Find out calcineurin-dependent functional genes in signaling pathways& Molecule transport cell wall maintenance and vesicular transport [67] | 40 |
| *Fch* | Cell cycle Alpha factor Fkh1&Fkh2 | Identify genes whose transcription is cell-cycle regulated [70] | 26 |
| *Met* | Metabolic reprogramming during diauxic shift | Identify genes affected by deletion of TUP1&YAP1 [9] | 7 |
| *Hyd* | Hydrostatic pressure | Transcript expression in yeast at high hydrostatic pressure [11] | 2 |
| *Iro* | Iron deprivation | Iron uptake experiment in yeast [53] | 6 |
| *Aft* | Aft2 iron regulation in yeast | Mutant containing a double aft1Deltaaft2Delta was generated to find overlapping Aft1 and Aft2 functions [48] | 2 |
| *Fit* | Iron uptake in yeast | Cells response to the absence of FIT genes by up-regulating systems of iron uptake [42] | 6 |
| *Pho* | Manipulation of phosphate levels | Identify components regulated by PHO regulatory pathway [40] | 8 |
| *Snf* | Snf-Swi mutants deleted | Find genes in vivo to remodel nucleosomes in vitro [59] | 12 |
| *Spo* | Sporulation of meiosis and spore morphogenesis | Identify genes at end of meiotic prophase [7] | 7 |

experiments as follows: *Alp* into 4 data sets {Alp1, Alp2, Alp3, Alp4}, *Des* into 7 data sets {Des1, Des2, Des3, Des4, Des5, Des6, Des7}, *Hea* into 20 data sets {Hea1, Hea1, Hea1, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15, Hea16, Hea17, Hea18, Hea19, Hea20}, *Dby* into 6 data sets {Dby1, Dby2, Dby3, Dby4, Dby5, Dby6}, *Cal* into 8 data sets {Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8}, and *Fch* into 2 data sets {Fch1, Fch2}, as detailed in Table 3.6.

These 47 partitioned data sets together with the remaining 10 unpartitioned data sets give us a total of 57 data sets of gene expression experiments based on the 57 experimental conditions (different sets of features on same set of gene)—$\mathcal{A}$ = {Alp1, Alp2, Alp3, Alp4, Ace, Des1, Des2, Des3, Des4, Des5, Des6, Des7, Haa, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15, Hea16, Hea17, Hea18, Hea19, Hea20, Dby1, Dby2, Dby3, Dby4, Dby5, Dby6, Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8, Fch1, Fch2, Met, Hyd, Iro, Aft, Fit, Pho, Snf, Spo}, as detailed in Table 3.7. The 57 datasets of yeast gene expression data has different objectives in finding genes in a specific functional group.

### 3.3.2 26 Functional Annotations of Yeast Genes

Mateos *et al.* [34] went further than studying the 5 cellular functions considered by Brown *et al.* [5]. In fact, they used the gene expression data sets of Eisen *et al.* [10] and tried to predict 96 functions of *S. cerevisiae* using multilayer perceptron. They reported that only 10% of functions are trainable by their approach—this is not surprising since many of the 96 functional classes have too few members or have ambiguous members. The Stanford Microarray Database [16] also possesses a huge collection of microarray data from global laboratories on 116 functions in *S. cerevisiae*

Table 3.6: Partition on 5 data sets into 45 data sets based on experiments.

| Old Dataset Code | Original Experiments factors | New Dataset code | Partitioned from original data set due to listed experimental factors | New data points |
|---|---|---|---|---|
| *Alp* | Alpha-factor based synchronization | Alp1 | Alpha-factor based synchronization | 18 |
| | | Alp2 | Elutriation | 14 |
| | | Alp3 | Cdc15 based synchronization | 24 |
| | | Alp4 | Cln3-Clb2 Experiments | 3 |
| *Des* | DES460 and 0.02%mms | Des1 | DES460 and 0.02% mms | 7 |
| | | Des2 | DES459 + 0.02% mec1 | 7 |
| | | Des3 | DUN1 + 0.02% MMS | 3 |
| | | Des4 | WT-plus-gamma | 8 |
| | | Des5 | DES460(wt)-mock irradiation | 4 |
| | | Des6 | mec1-plus-gamma | 8 |
| | | Des7 | DES459(mec1)-mock irradiation | 4 |
| *Hea* | Heat shock | Hea1 | Stress Response by Heat shock | 8 |
| | | Hea2 | Stress Response by SL3 | 5 |
| | | Hea3 | Stress Response by 25C | 5 |
| | | Hea4 | Stress Response by 29C to 33C | 4 |
| | | Hea5 | Stress Response by 29C+1M sorbitol to 33C+1M sorbitol | 3 |
| | | Hea6 | Stress Response by 33C no sorbitol | 2 |
| | | Hea7 | Stress Response by Heat shock upto 37 | 7 |
| | | Hea8 | Stress Response by constant 0.32mM.H202 redo | 10 |
| | | Hea9 | Stress Response by 1mM menadione redo | 9 |
| | | Hea10 | Stress Response by 2.5mMDTT | 8 |
| | | Hea11 | Stress Response by dtt | 7 |
| | | Hea12 | Stress Response by 1.5mMdiamide | 8 |
| | | Hea13 | Stress Response by 1M sorbitol | 6 |
| | | Hea14 | Stress Response by Hypo-osmotic shock | 5 |
| | | Hea15 | Stress Response by Amino Acid + Adenine starvation | 5 |
| | | Hea16 | Stress Response by Nitrogen Depletion | 9 |
| | | Hea17 | Stress Response by YPD 25C | 10 |
| | | Hea18 | Stress Response by YPD 30C | 9 |
| | | Hea19 | Stress Response by degree growth | 5 |
| | | Hea20 | Stress Response by Steady state | 8 |
| *Dby* | DBY8778 stationary phase | Dby1 | DBY8778 stationary phase | 6 |
| | | Dby2 | delYPL230W stationary phase | 6 |
| | | Dby3 | Wt-hypo-osmotic-shock | 6 |
| | | Dby4 | delPPT1-hypo-osmotic-shock | 5 |
| | | Dby5 | DBY8778-heat-shock | 5 |
| | | Dby6 | delKIN82-heat-shock | 5 |
| *Cal* | Calcium time series | Cal1 | Calcium time series | 8 |
| | | Cal2 | Calcium + FK506 time series | 8 |
| | | Cal3 | Calcium + FK506 vs calcium | 4 |
| | | Cal4 | Crz1+calcium vs CRZ1+calcium | 4 |
| | | Cal5 | Crz1+NaC1 vs CRZ1+NaC1 | 4 |
| | | Cal6 | NaC1 time series | 4 |
| | | Cal7 | NaC1 + FK506 vs NaC1 | 4 |
| | | Cal8 | NaC1 + FK50 time series | 4 |
| *Fch* | Cell cycle Alpha factor Fkh1&Fkh2 | Fch1 | Fkh1 Fkh2 Alpha Factor | 13 |
| | | Fch2 | Fkh1 Fkh2 Cell cycle Alpha factor | 13 |

Table 3.7: 57 microarray data sets used in our study.

| New Code | Experimental condition | Data Points |
|---|---|---|
| *Alp1* | Alpha-factor based synchronization | 18 |
| *Alp2* | Elutriation | 14 |
| *Alp3* | Cdc15 based synchronization | 24 |
| *Alp4* | Cln3-Clb2 Experiments | 3 |
| *Ace* | Transcriptional activators Ace1 and Mac1 | 6 |
| *Des1* | DES460 and 0.02% mms | 7 |
| *Des2* | DES459 + 0.02% mec1 | 7 |
| *Des3* | DUN1 + 0.02% MMS | 3 |
| *Des4* | WT-plus-gamma | 8 |
| *Des5* | DES460(wt)-mock irradiation | 4 |
| *Des6* | mec1-plus-gamma | 8 |
| *Des7* | DES459(mec1)-mock irradiation | 4 |
| *Haa* | Haa1 regulated transcription | 4 |
| *Hea1* | Stress Response by Heat shock | 8 |
| *Hea2* | Stress Response by SL3 | 5 |
| *Hea3* | Stress Response by 25C | 5 |
| *Hea4* | Stress Response by 29C to 33C | 4 |
| *Hea5* | Stress Response by 29C+1M sorbitol to 33C+1M sorbitol | 3 |
| *Hea6* | Stress Response by 33C no sorbitol | 2 |
| *Hea7* | Stress Response by Heat shock upto 37 | 7 |
| *Hea8* | Stress Response by constant 0.32mM.H202 redo | 10 |
| *Hea9* | Stress Response by 1mM menadione redo | 9 |
| *Hea10* | Stress Response by 2.5mMDTT | 8 |
| *Hea11* | Stress Response by dtt | 7 |
| *Hea12* | Stress Response by 1.5mMdiamide | 8 |
| *Hea13* | Stress Response by 1M sorbitol | 6 |
| *Hea14* | Stress Response by Hypo-osmotic shock | 5 |
| *Hea15* | Stress Response by Amino Acid + Adenine starvation | 5 |
| *Hea16* | Stress Response by Nitrogen Depletion | 9 |
| *Hea17* | Stress Response by YPD 25C | 10 |
| *Hea18* | Stress Response by YPD 30C | 9 |
| *Hea19* | Stress Response by degree growth | 5 |
| *Hea20* | Stress Response by Steady state | 8 |
| *Dby1* | DBY8778 stationary phase | 6 |
| *Dby2* | delYPL230W stationary phase | 6 |
| *Dby3* | Wt-hypo-osmotic-shock | 6 |
| *Dby4* | delPPT1-hypo-osmotic-shock | 5 |
| *Dby5* | DBY8778-heat-shock | 5 |
| *Dby6* | delKIN82-heat-shock | 5 |
| *Cal1* | Calcium time series | 8 |
| *Cal2* | Calcium + FK506 time series | 8 |
| *Cal3* | Calcium + FK506 vs calcium | 4 |
| *Cal4* | Crz1+Calcium vs CRZ1+calcium | 4 |
| *Cal5* | Crz1+NaC1 vs CRZ1+NaC1 | 4 |
| *Cal6* | NaC1 time series | 4 |
| *Cal7* | NaC1 + FK506 vs NaC1 | 4 |
| *Cal8* | NaC1 + FK50 time series | 4 |
| *Fch1* | Fkh1,Fkh2 Alpha Factor | 13 |
| *Fch2* | Fkh1,Fkh2 Cellcycle Alpha factor | 13 |
| *Met* | Metabolic reprogramming during diauxic shift | 7 |
| *Hyd* | Hydrostatic pressure | 2 |
| *Iro* | Iron deprivation | 6 |
| *Aft* | Aft2 iron regulation in yeast | 2 |
| *Fit* | Iron uptake in yeast | 6 |
| *Pho* | Manipulation of phosphate levels | 8 |
| *Snf* | SnfSwi mutants deleted | 12 |
| *Spo* | Sporulation of meiosis and spore morphogenesis | 7 |

proteins. These two data collections contain too many protein functions for us to conduct an exhaustive comparison between using multiple data sets vs using a single data set.

Functional classification of genes with 6 microarray datasets of yeast is studied by Brown [5] and Mateos [34], as briefed in our earlier Section 3.1.2. In this study, we take multiple datasets from different biological experiment with more functional annotations of yeast. Our study establishes the importance of dataset selection criteria. Munich Information Center for Protein Sequences (MIPS) [36] has latest functional annotation on yeast. 116 functions of *Saccharomyces cerevisiae* are taken, from MIPS catalogue (Version 2.0, dated 19-03-2004) at the **second level** (http://mips.gsf.de/projects/funcat). MIPS classification scheme is hierarchical with about 19 general classes at the **first level** and then further subdivided into more specific classes at the second level. Following earlier researchers [5, 34, 8] we also considered functions from the second level. After removing singleton function classes, our data set is reduced to 46 functions on 2114 genes. We segmented this data into 26 function (1928 genes) and 20 functions (186 genes) based on number of genes above 25 and less than 25, respectively. We report our findings on 26 function (as listed in Table 3.8) in this thesis report.

## 3.4 Algorithms and Methods

The algorithms used in this report are—naïve Bayesian [77, 78, 79], neural network [82, 83, 84], support vector machine [71, 72, 73], and decision tree C4.5 [74, 75, 76]. The feature selection methods used in this report are—Correlation-based feature subset selection [19], $\chi^2$ feature selection [32], and Information-gain feature

Table 3.8: 1928 yeast genes on 26 functional classes from MIPS.

| Original Code | Our code | Functions | Number of Annotated genes |
|---|---|---|---|
| 11.02 | Rsn | RNA-synthesis | 226 |
| 11.04 | Rpr | RNA-processing | 161 |
| 10.03 | Cyc | Cell-cycle | 149 |
| 20.09 | Trt | Transport-routes | 145 |
| 12.01 | Rib | Ribosome-biogenesis | 138 |
| 1.01 | Aam | Amino-acid-metabolism | 103 |
| 1.06 | Lim | Lipid-fatty-acid-and-isoprenoid-metabolism | 99 |
| 10.01 | Dna | DNA-processing | 99 |
| 1.05 | Ccm | C-compound-and-carbohydrate-metabolism | 82 |
| 1.03 | Nuc | Nucleotide-metabolism | 81 |
| 14.13 | Deg | Protein-degradation | 77 |
| 32.01 | Str | Stress-response | 58 |
| 1.07 | Vit | Metabolism-of-vitamins-cofactors-and-prosthetic-groups | 54 |
| 14.07 | Prm | Protein-modification | 48 |
| 20.01 | Tcs | Transported-compounds-(substrates) | 46 |
| 12.04 | Tra | Translation | 42 |
| 11 | Tcp | Transcription | 39 |
| 14.04 | Ptt | Protein-targeting-sorting-and-translocation | 37 |
| 34.11 | Csr | Cellular-sensing-and-response | 33 |
| 20.03 | Tfc | Transport-facilitation | 32 |
| 42.01 | Wal | Cell-wall | 32 |
| 12.10 | Ami | Aminoacyl-tRNA-synthetases | 31 |
| 43.01 | Fun | Fungal-micro-organismic cell type differentiation | 31 |
| 2.13 | Res | Respiration | 29 |
| 14.01 | Pfs | Protein-folding-and-stabilization | 29 |
| 32.07 | Dtx | Detoxification | 27 |

selection method.

**1-vs-All :** The positive class comprises those samples belonging to the function being studied; the rest of the samples in the data sets are treated as negative samples (1-vs-all scheme).

**WEKA package [65] :** Initially we used GIST package [5] on 5 functions of yeast to compare outcome with previous studies. Later, we used only WEKA package (as detailed below) on all the three research problems that we have studied.

1. WEKA (http://www.cs.waikato.ac.nz/ml/weka/)– Waikato Environment for Knowledge Analysis, Version 3-3-4 [65] from "Waikato University", New Zealand.

2. WEKA is installed on SunOS sparc, delta 5.8 server with 8GB RAM.

3. Perl scripts on SunOS are developed for automatic processing of different classification algorithms and feature selection methods in—WEKA.

4. In WEKA package while using $\chi^2$ feature selection method and Ranker method (weka.attributeSelection.Ranker) to evaluate the attribute, the "threshold" is set as $-T0$ to select attributes whose ranks values are above than "0".

5. In WEKA package while using Information-gain feature selection method and Ranker method (weka.attributeSelection.Ranker) to evaluate the attribute, the "threshold" is set as $-T0$ to select attributes whose ranks values are above than "0".

6. We use C4.5, Naïve Bayesian, multilayer perceptron (with one input layer and nodes=number of input attributes;one middle layer with nodes=(number of attributes+number of classes)/2;one output layer with nodes=number of classes),

and SVM (with polynomial kernel of degree 3) implementation from the WEKA package at its default setting.

**Evaluation metrics :** Machine learning algorithm builds a classification model based on training data and yields results on the given test data. The model is evaluated by checking the test data prediction ability of the model. We derive several units of classification—TP:Number of True Positives, FP:Number of False Positives, FN:Number of False Negatives, and TN:Number of True Negatives—based on the outcome on testing data. Evaluation metrics used to evaluate any classification model are listed below :

- Cost saving function, termed as $S(M,k) = k * TP - FP$ (for any k $> 0$ and for any machine learning algorithm "M"). The relative performance of different classifier is not affected by change of k value. A similar cost saving function was earlier suggested by Brown et al. [5].
- $F - measure = \frac{2*Recall*Precision}{Recall+Precision}$
- $Recall = \frac{TP}{TP+FN}$ and
- $Precision = \frac{TP}{TP+FP}$
- Rate of False Negative $= \frac{FN}{Numberof Positives}$, used by Mateos et al. [34].
- Rate of False Positive $= \frac{FP}{Numberof Negatives}$
- $Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$
- $Specificity = \frac{TN}{TN+FP}$

Table 3.9: ABREVIATIONS

| Abbreviation | Expansion of Abbreviation |
|---|---|
| SVM | Support Vector machine implementation of WEKA package |
| MLP | multilayer Perceptron |
| C4.5 | Decision tree C4.5 algorithm |
| NBay | Naïve Bayesian algorithm |
| CFS | Correlation-based feature selection method |
| Chi | $\chi^2$ feature selection method |
| Info | Information-gain feature selection method |
| Hill | Combination of Whole data sets by Hill climbing algorithm with selection one-data set-in each iteration |
| Greedy-Hill | Combination of Whole data sets by Greedy-Hill climbing algorithm with selection multiple-data set-in each iteration |
| ALL | Performance obtained in a classification model by using all available data sets |
| BI | Performance derived in a classification model by using the Best of Individual data set |
| Combination | Performance of a classification model with Combination of whole data sets selected by Hill or Greedy-Hill |
| Performance | Evaluation of a classification model by using cost saving measure $S(M, 2)$ |
| unrelated | Different sets of features on same set of genes or types of sites |
| FS | Performance obtained in a classification model by using features selected by feature selection methods |
| Feat | Features that are selected by feature selection methods |
| $S(M, k)$ | Value by cost saving measure for any k $> 0$ and any machine learning algorithm (m) |
| $F(M)$ | Value by F measure by any machine learning algorithm (m) |
| 3-fold | Data used in the classification model is by 3-fold cross validation scheme |
| Rt F N | Rate of False Negative |
| Rt F P | Rate of False Positive |

- $Sensitivity = \frac{TP}{TP+FN}$

We analysed yeast catalogue Version 2.0 (dated 19-March-2004, used in our study) and Version 2.1 (dated 9-January-2007). Some changes in functional annotations at level-2 are noted and listed in Table 3.10. Nevertheless our data and functional annotations are good enough as of today, even after going through the new catalogue.

Table 3.10: Updated functional annotations as per Version 2.1 yeast catalogue.

| Updated function | Cat-Code | Our code | Functions | Genes |
|---|---|---|---|---|
| nucleotide-nucleoside-nucleobase metabolism | 1.03 | Nuc | Nucleotide-metabolism | 161 |
| Protein Peptide degradation | 14.13 | Deg | Protein-degradation | 29 |
| Transport facilities | 20.03 | Tfc | Transport-facilitation | 27 |
| cellular sensing and response to external stimulus | 34.11 | Csr | Cellular-sensing-and-response | 29 |
| Nitrogen, sulfur and selenium metabolism | 1.02 | Nsm | Nitrogen-and-sulfur-metabolism | 20 |
| cellular signalling | 30.01 | Int | Intracellular-signalling | 7 |
| homeostasis | 34.01 | Hom | Ionic-homeostasis | 6 |
| cytoskeleton-structural proteins | 42.04 | Cyt | Cytoskeleton | 3 |

# Chapter 4

# Exploring Existing Methods

We demonstrate in this chapter that judicious use of additional data sets—even those that are derived from very different wet experimental conditions (different sets of features on same set of genes or sites)—can increase the accuracy of classification models for a variety of bioinformatics prediction problems. In particular, we provide three substantial examples to demonstrate this point. In the first two examples where exhaustive comparisons are possible, we show that using additional data sets (on same set of genes or sites) can yield better results than using a single data set with 26% and 31% chance respectively. In the third example, the number of combinations is too large for us to compare exhaustively, but we are able to find some combinations of additional data sets that produce better results than any single data set. This seems to suggest that using all available data sets may be a simple way to improve prediction accuracy. However, in this chapter, we show how using all available data sets does not give the best improved prediction accuracy and often gives a worse accuracy than using the best individual data sets. In the first two problems, where exhaustive comparisons are possible, we show that there is a 26% and 11% chance respectively that using a combination of 2 or more data sets, but not all of the available data

sets, yields better prediction performance by $S(M, 2)$ than using all of the available data sets. In the third example, where exhaustive comparisons are not feasible due to the large number of data sets, we show for 79% of the protein functional classes, we are able to use a combination of 2 or more data sets, but not all of the available data sets to obtain a higher prediction performance by $S(M, 2)$ that using all the available data sets. The next choice generally used in classification problems by many researchers is to apply feature selection methods and build a model from the selected features. We illustrate, in this chapter, how prediction accuracy can be improved by using conventional feature selection methods compared to using the best individual data sets or all available data sets. However, we also show that conventional feature selection methods do not achieve the best prediction accuracy often enough than using a combination of whole individual data sets.

Specifically:

- Eisen *et al.* [10] showed that genes involved in common functional activities would have similar gene expression profiles under a number of experimental conditions. Brown *et al.* [5] studied—in a follow-up to Eisen *et al.*—how 5 specific cellular functions of *S. cerevisiae* are affected under 6 experimental conditions. This resulted in 6 gene expression data sets described earlier in Table 3.1. These data sets can be used, either individually or in combination to infer if a gene is involved in the 5 specific cellular functions studied. Brown *et al.* [5] augmented the data sets of Eisen et al with 2 additional experimental data sets to infer 5 specific functions of *S. cerevisiae* genes. Mateos *et al.* [34] used the 6 expression data sets of Eisen et al to also infer the same 5 specific cellular functions of *S. cerevisiae* genes. The point to note here is all of them used all data sets that

are conveniently available in their classification models. We show here that we can much more accurately infer if a gene is involved in the 5 specific cellular functions, if we use these 6 data sets in combination as opposed to using any single one of them. Our results show that using multiple data sets in combination has as much as 99% (74%, 15%, respectively) chance of yielding better classification accuracy than using any single data set for inferring ribosomal (proteasome, TCA-cycle, respectively) proteins by SVM. Experiments on other classifiers—such as Naïive Bayesian and C4.5—also show large probability of yielding higher accuracy for inferring ribosomal, proteasome, and TCA-cycle proteins, when multiple data sets are used than when any single data set is used. In fact, out of the 4940 exhaustive comparisons made over SVM, C4.5, Naïve Bayesian, and MLP on the 5 specific cellular functions, 258 (= 5%) of the possible combination of data sets yield an accuracy equal to the best of all individual data sets, and 1266 (= 26%) of the possible combination of data sets yield better accuracy than the best of all individual data sets. That is, using multiple data sets has 26% chance of yielding better results. Next, we show that we can infer more accurately if a gene is involved in the 5 specific cellular functions, when we use some combination of data sets but not necessarily all the available data sets. Our results show that using a combination of 2 or more data sets—but not all the available data sets—has as much as 98% (68%, respectively) chance of yielding better classification accuracy than using all available data sets for respiration (ribosomal, respectively) proteins by MLP. Studies on other classifiers—SVM, C4.5, and NBay—also show large probability of yielding higher accuracy for inferring respiration and ribosomal,

when 2 or more data sets but not all available data sets are used. In fact, out of the 4920 exhaustive comparisons over SVM, C4.5, NBay, and MLP on the 5 specific cellular functions, 81 (=2%) of the possible combinations of data sets yield an accuracy equal to using all available data sets, and 1258 (=26%) of the possible combinations of data sets yield better accuracy than using all available data sets. Finally, we show here that feature selection methods can yield better results than using the best individual data sets or using all available data sets. Results show that, for 80% (60%, 60%, and 60%, respectively) of the protein functional classes, we are able to use feature selection methods to obtain a higher prediction accuracy than using the best of individual data sets through C4.5 (SVM, NBay, and MLP, respectively). Results also show that, for 80% (60%, 80%, 80%, respectively) of the protein functional classes, we are able to use feature selection methods to obtain a higher prediction accuracy than using all available data sets through C4.5 (SVM, NBay, and MLP, respectively). Finally we show for 60% (60%, 80%, and 80%) of the protein functional classes, we are able to use a combination of 2 or more whole data sets to obtain a higher prediction accuracy than using the best performance by $S(M, 2)$ from conventional feature selection methods, through C4.5 (SVM, NBay, and MLP, respectively). That is, while the conventional feature selection approach is a significant improvement over the use of the best of individual data sets and over the use of all data sets blindly, it does not lead to the best accuracy often enough for protein functional classification problem. To the best of our knowledge, the only significant previous work on inferring these 5 specific cellular functions, based on these 6 data sets, are that of Brown *et al.* [5] and Mateos *et al.* [34]. Both of

them obtained classification accuracy similar to ours, though the numbers cannot be directly compared due to certain differences in the treatment of the data sets between them and us. Nevertheless, both Brown *et al.* and Mateos *et al.* used all available data sets, and did not consider the issue of using single data sets vs using multiple data sets.

- Protein sites are micro-environments within a biomolecular structure, distinguished by their structural or functional role. Bagley *et al.* [57] formulated 6 sets of micro-environment properties—comprising co-ordinates, chemical groups, secondary structures, atom-based, residue-based, and others properties—surrounding a protein site. They showed that the distributions of these properties differ significantly between sites and non-sites. In particular, Bagley *et al.* studied calcium binding sites, serine protease active sites, and disulfide bridges. Subsequently, Wei *et al.* [31] built a classification model on calcium binding site using the 19 micro-environment properties formulated by Bagley *et al.* [57] as features. Wei *et al.* took 16 calcium binding sites and 100 non-binding sites, and built a Bayesian classification model using 5 data sets formulated by Bagley *et al.* [57], except co-ordinates. The point to note is that Wei *et al.* used 5 categories of micro-environment properties in inferring calcium binding sites. Again, these 6 sets of micro-environment properties are very different from each other in nature. They can be used either individually or in combination to infer if a candidate region in a protein is a calcium binding site. We show here that we can much more accurately infer if a candidate region is a calcium binding site, a serine protease active site, or a disulfide bridge, using multiple sets of micro-environment properties than using any single set of micro-environment

properties. Our results show that using multiple sets of micro-environment properties in combination gives a 81% (79%, 89%, respectively) chance of more accurately inferring a candidate site as a calcium binding site (serine protease active site, disulfide bridge, respectively), than using any single best set of micro-environment properties by SVM. Experiments on other classifiers such as Naïive Bayesian, C4.5, and MLP also show good gain in accuracy when multiple sets of micro-environment properties are used. In fact, out of 684 exhaustive comparisons made over SVM, Naïve Bayesian, C4.5, and MLP on 6 sets of micro-environment properties on the 3 types of protein sites, 13 (= 2%) of the possible combinations of sets of micro-environment properties yield an accuracy equal to the best of any single set of micro-environment properties, and 213 (= 31%) yield better accuracy. That is, using multiple sets of micro-environment properties has 31% chance of yielding better results. Next, we show here we can much more accurately infer if a candidate region is a calcium binding site, a serine protease active site, or a disulfide bridge, using a combination of 2 or more of micro-environment properties—but not all available data sets—than using all available data sets of micro-environment properties. Our results show that using multiple sets of micro-environment properties in combination gives a 33% (7%, respectively) chance of more accurately inferring a candidate site as a calcium binding site (serine protease active site, respectively), than using all available data sets of micro-environment properties by SVM. Experiments on other classifiers such as NBay, C4.5, and MLP also show gain in accuracy when multiple sets of micro-environment properties are used. In fact, out of the 684 exhaustive comparisons made over SVM, C4.5, NBay, and MLP on the 3

types of protein sites, 12 (=2%) of the possible combinations of sets of micro-environment properties yield an accuracy equal to using all available data sets of micro-environment properties, and 71 (=10%) of the possible combinations of sets of micro-environment properties yield better accuracy. Finally, we show here that feature selection methods can yield better results than using the best of the individual sets of micro-environment properties or using all sets of micro-environment properties. Results show that for 100% (100%, 67% and 100%, respectively) of the types of protein sites, we are able to use feature selection methods to obtain a higher prediction accuracy than using the best individual sets of micro-environment properties through C4.5 (SVM, NBay, and MLP, respectively). Results also show that for 67% (67%, 33%, 67%, respectively) of the types of protein sites, we are able to use feature selection methods to obtain a higher prediction accuracy than using all 6 sets of micro-environment properties through C4.5 (SVM, NBay, and MLP, respectively). Finally we show for 100% (100%, 100%, and 67%) of the types of protein sites, we are able to use a combination of 2 or more sets of micro-environment properties to obtain a higher prediction accuracy than using the best performance by $S(M, 2)$ from feature selection methods, through C4.5 (SVM, NBay, and MLP, respectively). Thus, while the conventional feature selection approach is a significant improvement over the use of the best of individual data sets and over the use of all data sets blindly, it does not lead to the best accuracy often enough for this protein site classification problem. To the best of our knowledge, Bagley *et al.* [57] and Wei *et al.* [31] have one of the best results in classifying calcium binding sites, serine protease active sites, and disulfide bridges. Their reported accuracies are

similar to ours, though the numbers cannot be directly compared due to differences in data sets used—we are unable to obtain their data sets. Nevertheless, both Bagley *et al.* and Wei *et al.* used all available sets of micro-environment properties data, and did not consider the issue of using single data sets vs using multiple data sets.

- Mateos *et al.* [34] went further than studying the 5 cellular functions considered by Brown *et al.* [5]. In fact, they used the gene expression data sets of Eisen *et al.* [10] and tried to predict 96 functions of *S. cerevisiae* using multilayer perceptron. They reported that only 10% of functions are trainable by their approach—this is not surprising since many of the 96 functional classes have too few members or have ambiguous members. The Stanford Microarray Database [16] also possesses a huge collection of microarray data from global laboratories on 116 functions in *S. cerevisiae* proteins. These two data collections contain too many protein functions for us to conduct an exhaustive comparison between using multiple data sets vs using a single data set, using all available data sets, and using features selected by feature selection methods. Nevertheless, we consider 26 functions of *S. cerevisiae* and 16 data sets with different experimental conditions from the Stanford Microarray Database. These 26 functional classes are chosen because they have at least 3 unambiguous member genes. We show that for most of the 26 functional classes, we can find a combination of data sets from the 16 different experimental conditions that yield better accuracy than using the best of all single data sets. Next, we show that for most of the 26 functional classes, we can find a combination of data sets from the 16 different experimental conditions that yield better accuracy

than using all the 16 data sets together. We show here that feature selection methods can yield better results than using the best individual data sets or using all available data sets. Results show that for 37% (24%, 70%, and 43%, respectively) of the protein functional classes, the use of conventional feature selection methods lead to a poorer prediction accuracy than using the best of individual data sets through C4.5 (SVM, NBay and MLP, respectively). For 57% (65%, 30%, and 43%, respectively) of the protein functional classes, it lead to an equal prediction accuracy than using the best of individual data sets through C4.5 (SVM, NBay and MLP, respectively). Results show that for 63% (74%, 89%, and 46%, respectively) of the protein functional classes, we are able to use feature selection methods to obtain a higher prediction accuracy than using all available data sets through C4.5 (SVM, NBay, and MLP, respectively). Finally, we show for at least 37% (43%, 72%, and 61%, respectively) of the protein functional classes, we are able to use a combination of 2 or more data sets to obtain a higher prediction accuracy than using the best performance by $S(M, 2)$ from feature selection methods, through C4.5 (SVM, NBay, and MLP, respectively). So, while the feature selection approach is a significant improvement over the use of the best of individual data sets and over the use of all data sets blindly, it does not lead to the best accuracy often enough for this protein functional classification problem.

## 4.1 Using Best Individual Data Set

In this section we illustrate modeling of 5 specific cellular functions of yeast genes and 26 functions of yeast by using only the best of microarray data sets (out of 6 and

57 data sets, respectively). Also we show modeling on 3 types of protein sites using the best of micro-environment property (out of 6 categories).

### 4.1.1 Use of Best Microarray Data Set on 5 Functions of Yeast Genes

In Section 1.3.1 we detailed earlier studies on modeling five functions of yeast using six microarray data sets. More details about the 5 cellular functions have already been described in Table 3.2.

We now take the same 6 sets of gene expression experiments from the paper of Eisen *et al.*—based on the 6 experimental conditions (different sets of features on same set of genes)—as individual data sets, as depicted in Table 3.1. Then for a function $f$, for a learning method $m$, and for a gene expression data set $C$, we construct a classifier $C_f(C, m)$ as follows. We take those annotated genes from the study of Brown *et al.* that have function $f$ to be our positives, and those that do not have function $f$ as negatives. Then for each positive gene, we build its corresponding feature vector by taking its gene expression values from the data set $C$. Similarly, for each negative gene, we build its corresponding feature vector by taking its gene expression values from the data set $C$. Then we apply the learning method $m$ to these feature vectors and obtain the performance measure $S(M, 2)$ by 3-fold cross validation.

We show below in Table 4.1 the results of the experiments just described, using SVM as the learning method, and each of the gene expression results from the 6 experimental (different sets of features on same set of genes) conditions as a data set. We can think of this table as the performance by $S(M, 2)$ of SVM for predicting 5 specific cellular functions of yeast genes using an *individual* data set based on a specific experimental condition. The rows are the 5 functions—TCA (TCA cycle),

RESP (respiration), RIBO (ribosomes), PROT (proteosomes), and HIST (histones). The second through the seventh columns are the 6 data sets based on 6 experimental conditions (different sets of features on same set of genes)—Alp ($\alpha$-factor arrest), Cdc (cdc15 arrest), Elu (Elutriation), Spo (Sporulation), Dia (Diauxic shift), Ccc (Cln3 and Clb2). The SVM here is the support vector machine implementation from the GIST package and uses RBF of degree 3, as in Brown *et al.* Note that the each feature vector derived from the Ccc data set has only 3 feature points, as there are only 3 Ccc experiments; see Table 3.1. Note also that Eisen *et al.* did only a clustering of genes based on their gene expression profiles from the 6 experimental conditions, and did not annotate the genes with explicit function. Here we annotate the genes using information from the MIPS Catalogue (Version 1.3) dated 25th June 2003.

Table 4.1: Performance by $S(M, 2)$ on 5 functions of yeast based on individual data set through SVM.

| Function | Learning cost savings $S(SVM, 2)$ | | | | | |
|---|---|---|---|---|---|---|
| | *Alp* | *Cdc* | *Elu* | *Spo* | *Dia* | *Ccc* |
| TCA | -360 | -5 | 0 | -157 | -532 | -661 |
| RESP | -232 | -160 | -258 | -348 | -1319 | -761 |
| RIBO | -250 | 69 | -66 | -6 | -612 | -1347 |
| PROT | -438 | -66 | -367 | -27 | -116 | -398 |
| HIST | 2 | 16 | -2 | 11 | -87 | -125 |

We also show below in Table 4.2 the results of the experiments, using MLP as the learning method, and each of the gene expression results from the 6 experimental conditions (different sets of features on same set of genes) as a data set. We can think of this table as the performance by $S(M, 2)$ of MLP for predicting 5 specific cellular functions of yeast genes using an *individual* data set based on a specific experimental

condition. The MLP is the multilayer perceptron implementation from the WEKA package at its default setting.

Table 4.2: Performance by $S(M, 2)$ on 5 functions of yeast based on individual data set through MLP.

| Function | Learning cost savings $S(MLP, 2)$ | | | | | |
|---|---|---|---|---|---|---|
| | *Alp* | *Cdc* | *Elu* | *Spo* | *Dia* | *Ccc* |
| TCA | -6 | -6 | -11 | -1 | -1 | 0 |
| RESP | 0 | -4 | 0 | 0 | -1 | 0 |
| RIBO | 117 | 117 | 138 | 182 | 174 | 6 |
| PROT | -6 | -2 | -3 | 24 | 3 | 0 |
| HIST | 14 | 16 | 18 | 10 | 0 | 0 |

Spellman *et al.* [56] suggested that some data points in their study could be removed as their appeared to be aberrant. While Eisen *et al.* used the data of Spellman *et al.* in full, Brown *et al.* [5] removed 15 time points. Specifically, Brown *et al.* removed 10 time points from the CDC data set, all 3 time points from the CCC data set, and 2 time points from the SPO data sets. At the same time, Brown *et al.* [5] also added 14 new time points from 3 new experimental conditions. Specifically, Brown *et al.* added 6 time points from a data set labeled HEAT, 4 time points from a data set labeled DTT, and 4 time points from a data set labeled COLD, provided on the website (http://www.cse.ucsc.edu/research/compbio/genex). Note that Brown *et al.* [5] annotated the genes using the MIPS Catalogue of 8th November 2001. Thus they have 17 TCA genes, 30 RESP genes, 121 RIBO genes, 35 PROT genes, and 11 HIST genes. However, here we annotate the genes using information from the MIPS Catalogue (Version 1.3) dated 25th June 2003 which is more up to date; see Table 3.2.

We show below in Table 4.3 the results of $C_f(C, m)$, for various learning methods $m \in \{$ C4.5, SVM, NBay, MLP$\}$, for 5 functions $f \in \{$HIST, PROT, RESP, RIBO, TCA$\}$ annotated as per MIPS Catalogue (Version 1.3) dated 25th June 2003, and for various data sets $C \in \{$Alp, Cdc, Elu, Spo, Dia, HEAT, DTT, COLD$\}$ derived from experimental conditions (different sets of features on same set of genes) as per Brown *et al.* In this table, we show the $S(M, 2)$ score based on the performance by $S(M, 2)$ of $m$ on the best individual data set; that is, for a function $f$, $S(M, 2) = \max_{C \in \{\text{Alp,Cdc,Elu,Spo,Dia,HEAT,DTT,COLD}\}} C_f(C, m)$.

Table 4.3: Performance by $S(M, 2)$ on 5 functions of yeast based on the best of individual data sets through algorithms.

| Function | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|
| HIST | 12 | 16 | 13 | 16 |
| PROT | 15 | 0 | 1 | 31 |
| RESP | 0 | 0 | 0 | 0 |
| RIBO | 174 | 160 | 174 | 189 |
| TCA | 0 | 0 | 0 | 3 |

### 4.1.2 Use of Best Micro-Environment Property on 3 Types of Protein Sites

We now take the same 6 categories of micro-environment properties as our 6 individual data sets, as depicted in Table 3.3. Then for a protein site $s$, for a learning method $m$, and for a data set $D$, we construct a classifier $C_s(D, m)$ as follows. We take atoms that are protein site $s$ to be our positives, and those that are not protein site $s$ as negatives. Then for each positive atom, we build its corresponding feature vector by taking micro-environment values from the data set $D$. Similarly, for each negative atom, we build its corresponding feature vector by taking micro-environment values

from the data set $D$. Then we apply the learning method $m$ to these feature vectors and obtained the performance measure $S(M, 2)$ by 3-fold cross validation.

We show below in Table 4.4 the results of the experiments just described, using SVM as the learning method, and each of the data sets results from the 6 categories of micro-environment properties. We can think of this table as the performance by $S(M, 2)$ of SVM for predicting 3 specific types of protein sites using an *individual* data set based on a specific category. The rows are the 3 types of protein sites—CALCIUM (calcium binding sites), DISULFIDE (disulfide bridges) and SERINE (serine protease active sites). The second through the seventh columns are the 6 data sets based on 6 category of micro-environment properties—Atom (4 atomic types and charges), Chemical (6 chemical-based properties), Residue (3 residue type classifications), Sec-str (2 secondary structure classifications), Others (4 general properties and measures), and Co-ord (3 dimensional co-ordinates of atoms). The SVM here is the Support Vector machine implementation from the WEKA package with polynomial kernel of degree 3.

Table 4.4: Performance by $S(M, 2)$ on 3 types of protein sites based on individual micro-environment property through SVM.

| | $S(SVM, 2)$ | | | | | |
|---|---|---|---|---|---|---|
| Prot-site | *Atom* | *Chemical* | *Residue* | *Sec-str* | *Others* | *Co-ord* |
| CALCIUM | 93 | 76 | 127 | 78 | 112 | 123 |
| SERINE | 0 | 0 | 0 | 0 | 0 | 15 |
| DISULFIDE | 0 | 0 | 0 | 0 | 1 | 6 |

We also show below in Table 4.5 the results of the experiments, using MLP as the learning method, and each of the 6 categories of micro-environment properties as a data set. We can think of this table as the performance by $S(M, 2)$ of MLP for

predicting 3 types of protein sites using an *individual* data set based on a specific category. The MLP is the multilayer perceptron implementation from the WEKA package at its default setting.

Table 4.5: Performance by $S(M, 2)$ on 3 types of protein sites based on individual micro-environment property through MLP.

| | Learning cost savings $S(MLP, 2)$ | | | | | |
|---|---|---|---|---|---|---|
| Prot-site | *Atom* | *Chemical* | *Residue* | *Sec-str* | *Others* | *Co-ord* |
| CALCIUM | 66 | 80 | 132 | 148 | 107 | 171 |
| SERINE | -6 | -2 | 7 | 80 | -7 | 73 |
| DISULFIDE | 0 | 0 | 43 | 0 | 13 | 64 |

We show below in Table 4.6 the results of $C_s(D, m)$, for various methods $m \in$ {C4.5, SVM, NBay, MLP}, for 3 types of protein sites $s \in$ {CALCIUM, DISULFIDE, SERINE}, and for various data sets $D \in$ {Atom, Chemical, Residue, Sec-str, Others, Co-ord} derived from categories of micro-environment properties. We show the $S(M, 2)$ score based on the performance by $S(M, 2)$ of $m$ on the best individual data set; that is,

for a site $s$, $S(M, 2) = \max_{D \in \{\text{Atom,Chemical,Residue,Sec-str,Others,Co-ord}\}} C_s(D, m)$.

Table 4.6: Performance by $S(M, 2)$ on 3 types of protein sites based on the best of individual micro-environment properties through algorithms.

| Prot-site | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|
| CALCIUM | 179 | 127 | 136 | 171 |
| SERINE | 80 | 15 | 72 | 80 |
| DISULFIDE | 58 | 6 | 53 | 64 |

### 4.1.3 Use of Best Microarray Data Set on 26 Functions of Yeast Genes

Section 3.3.1 shows 16 data sets that are retrieved from SMD (in Table 3.5) and 57 data sets that are derived after partition (in Table 3.7). Section 3.3.2 shows details on 26 functional annotations of yeast genes (in Table 3.8) for which we illustrate various outcomes.

Now, we have 57 data sets (47 partitioned and 10 unpartitioned data sets) on 26 functions of *S. cerevisiae*. We have 392 time points (for each experiment) on 2114 functionally known genes from the 57 data sets. We take the set of 57 gene expression experiments—based on the 57 experimental conditions (different sets of features on same set of genes)—as individual data sets, as depicted in Table 3.7. Then for a function $f$, for a machine learning method $m$, and for a gene expression data set $E$, we construct a classifier $C_f(E, m)$ as follows. We take those annotated genes that have function $f$ to be our positives, and those that do not have function $f$ as negatives. Then for each positive gene, we build its corresponding feature vector by taking its gene expression values from the data set $E$. Similarly, for each negative gene, we build its corresponding feature vector by taking its gene expression values from the data set $E$. Then we apply the learning method $m$ to these feature vectors and obtain the performance measure $S(M, 2)$ by 3-fold cross validation.

Table 4.7 shows one table of results (out of six tables) for each set of 26 functions, from the experiments we just described using C4.5 as the learning method, and each of the gene expression results from the 12 experimental conditions (different sets of features on same set of genes) as a data set (we are showing only outcome from 12 out of 57 data sets on 26 functions due to space constraint).

We can think of these tables as the performance by $S(M, 2)$ of C4.5 for predicting 26 cellular functions of yeast genes using an *individual* data set based on a specific experimental condition. The rows are the 26 functions (please refer to Table 3.8) shown in column 1 as catalogue number and column 2 as function code. Column 3 shows the number of genes for each function. The fourth through the fifteenth columns are the 12 data sets based on 12 experimental conditions (different sets of features on same set of genes)—*Alp*1 (Alpha-factor based synchronization), *Ace* (Transcriptional activators Ace1 and Mac1), *Alp*4 (Cln3-Clb2 Experiments), *Des*1 (DES460 and 0.02% mms), *Des*2 (DES459 + 0.02% mec1), *Des*3 (DUN1 + 0.02% MMS), *Des*4 (WT-plus-gamma), *Des*5 (DES460(wt)-mock irradiation), *Des*6 (mec1-plus-gamma), *Des*7 (DES459(mec1)-mock irradiation), *Alp*2 (Elutriation), and *Haa* (Haa1 regulated transcription). The C4.5 here is the decision tree algorithm C4.5 implementation from the WEKA package. Here we annotate the genes using information from the MIPS Catalogue (Version 2.0) of 19 March 2004.

We show in Table 4.8 the results of $C_f(E, m)$ for 26 functions, where $m \in \{$C4.5, SVM, NBay, MLP$\}$, for 26 functions $f \in \{$Aam, Nsm, Nuc, Pho, Ccm, Lim, Vit, Tca, Res, Fer, Mer, Ecr, Dna, Cyc, Tcp, Rsn, Rpr, Rmo, Pro, Rib, Tra, Trc, Ami, Pft, Pfs, Ptt, Prm, Apc, Deg, Tcs, Tfc, Trt, Int, Rdv, Str, Dtx, Hom, Csr, Tvp, Gro, Dea, Wal, Cyt, Nuc, Mit, Fun$\}$ annotated as per MIPS Catalogue (Version 2.0) of 19th March 2004, and for various data sets $E \in \mathcal{E} = \{$Alp1, Alp2, Alp3, Alp4, Ace, Des1, Des2, Des3, Des4, Des5, Des6, Des7, Haa, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15, Hea16, Hea17, Hea18, Hea19, Hea20, Dby1, Dby2, Dby3, Dby4, Dby5, Dby6, Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8, Fch1, Fch2, Met, Hyd, Iro, Aft, Fit, Pho, Snf, Spo$\}$ derived

Table 4.7: Performance by $S(M, 2)$ on 26 functions of yeast based on 12 individual data set through C4.5.

| Function | Code | Genes | Alp1 | Ace | Alp4 | Des1 | Des2 | Des3 | Des4 | Des5 | Des6 | Des7 | Alp2 | Haa |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | -2 | 0 | 0 | -6 | -8 | 0 | -3 | 0 | 0 | 0 | 0 | 0 |
| 11.04 | Rpr | 161 | 1 | 0 | 0 | -12 | 0 | 0 | 4 | 0 | 8 | 0 | -1 | 0 |
| 10.03 | Cyc | 149 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.09 | Trt | 145 | -4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12.01 | Rib | 138 | 79 | 71 | 9 | 74 | 73 | 0 | 131 | 56 | 99 | 94 | 100 | 0 |
| 1.01 | Aam | 103 | -3 | 0 | 0 | 16 | 3 | 26 | 0 | 0 | 2 | 0 | -4 | 0 |
| 1.06 | Lim | 99 | -7 | 0 | 0 | 3 | -1 | -3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10.01 | Dna | 99 | -4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -9 | 0 |
| 1.05 | Ccm | 82 | 0 | 0 | 0 | -2 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 |
| 1.03 | Nuc | 81 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 14.13 | Deg | 77 | -11 | 0 | 0 | -2 | 0 | 0 | -3 | 0 | 0 | 0 | -4 | 0 |
| 32.01 | Str | 58 | -6 | 0 | 0 | 0 | -8 | 0 | 0 | 0 | 0 | 0 | -2 | 0 |
| 1.07 | Vit | 54 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.07 | Prm | 48 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 |
| 12.04 | Tra | 42 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | Tcp | 39 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.03 | Tfc | 32 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 42.01 | Wal | 32 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12.10 | Ami | 31 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 43.01 | Fun | 31 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2.13 | Res | 29 | 0 | 0 | 0 | 0 | 3 | -2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.01 | Pfs | 29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32.07 | Dtx | 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

from experimental conditions (different sets of features on same set of genes). In these tables, we show the $S(M, 2)$ score based on the performance by $S(M, 2)$ of $m$ on the best individual data set; that is, for a function $f$, $S(M, 2) = \max_{E \in \mathcal{E}} C_f(E, m)$.

Table 4.8: Performance by $S(M, 2)$ on 26 functions of yeast based on the best of individual data sets through algorithms.

| Function | Code | Genes | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | 5 | 16 | 1 | 6 |
| 11.04 | Rpr | 161 | 8 | 0 | 3 | 7 |
| 10.03 | Cyc | 149 | 4 | 0 | 1 | 2 |
| 20.09 | Trt | 145 | 0 | 0 | 2 | 0 |
| 12.01 | Rib | 138 | 213 | 214 | 144 | 217 |
| 1.01 | Aam | 103 | 30 | 11 | 11 | 38 |
| 1.06 | Lim | 99 | 3 | 0 | 0 | 4 |
| 10.01 | Dna | 99 | 6 | 0 | 0 | 13 |
| 1.05 | Ccm | 82 | 1 | 2 | 0 | 4 |
| 1.03 | Nuc | 81 | 1 | 0 | 0 | 7 |
| 14.13 | Deg | 77 | 0 | 0 | 5 | 10 |
| 32.01 | Str | 58 | 1 | 4 | 0 | 2 |
| 1.07 | Vit | 54 | 0 | 0 | 0 | 0 |
| 14.07 | Prm | 48 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | 0 | 0 | 0 | 2 |
| 12.04 | Tra | 42 | 0 | 0 | 0 | 0 |
| 11 | Tcp | 39 | 0 | 0 | 0 | 0 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | 5 | 6 | 0 | 9 |
| 20.03 | Tfc | 32 | 0 | 0 | 0 | 0 |
| 42.01 | Wal | 32 | 0 | 0 | 0 | 0 |
| 12.10 | Ami | 31 | 3 | 0 | 0 | 4 |
| 43.01 | Fun | 31 | 0 | 0 | 0 | 1 |
| 2.13 | Res | 29 | 3 | 0 | 0 | 8 |
| 14.01 | Pfs | 29 | 0 | 0 | 0 | 0 |
| 32.07 | Dtx | 27 | 0 | 2 | 0 | 4 |

## 4.2 Using Additional Data Set

We show here that we can much more accurately infer if a gene is involved in the 5 specific cellular functions, if we use these 6 data sets in combination as opposed to using any single one of them. Our results show that using multiple data sets in combination has as much as 99% (74%, 15%, respectively) chance of yielding better classification accuracy than using any single data set for inferring ribosomal

(proteasome, TCA-cycle, respectively) proteins by SVM. We show here that we can much more accurately infer if a candidate region is a calcium binding site, a serine protease active site, or a disulfide bridge, using multiple sets of micro-environment properties than using any single set of micro-environment properties. Our results show that using multiple sets of micro-environment properties in combination gives a 81% (79%, 89%, respectively) chance of more accurately inferring a candidate site as a calcium binding site (serine protease active site, disulfide bridge, respectively), than using any single best set of micro-environment properties by SVM. We show that for most of the 26 functional classes, we can find a combination of data sets from the 16 different experimental conditions that yield better accuracy than using the best of all single data sets.

### 4.2.1 Use of Additional Microarray Data Set on 5 Functions of Yeast Genes

In Subsection 4.1.1, we provided the accuracy of inferring if a gene has one of the 5 specific functions—HIST, PROT, RESP, RIBO, or TCA—based on a feature vector derived from gene expression profile of that gene under one experimental condition. It is natural to speculate if one can perform better by using a feature vector derived by combining or merging gene expression profiles of that gene under two or more experimental conditions (different sets of features on same set of genes). That is, does the use of additional data improve classification accuracy for these 5 cellular functions? Recall that for data set of Brown *et al.*, with genes annotated as per MIPS Catalogue (Version 1.3) of 25th June 2003, there are 8 data sets corresponding to 8 experimental conditions (different sets of features on same set of genes). This yields 247 ($= 2^8 - 9$) non-empty combinations that involved more than 1 of these

8 data sets. This is a sufficiently small number of combinations. So we are able to consider the performance by $S(M, 2)$ of 4 learning methods $m \in \{$C4.5, SVM, NBay, and MLP$\}$ for predicting the 5 specific functions $f \in \{$HIST, PROT, RESP, RIBO, TCA$\}$ using these 247 combinations of data sets, and compare the results with using the 8 data sets individually.

The results from this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 4.9, 4.11, 4.10, and 4.12. In each of these 4 tables, the second column shows $|\{C_f(C, m) < k_f |\ C \in \mathcal{C}\}|$, the third column shows $|\{C_f(C, m) = k_f |\ C \in \mathcal{C}\}|$, and the fourth column shows $|\{C_f(C, m) > k_f |\ C \in \mathcal{C}\}|$, where $k_f = \max_{C \in \{\text{Alp,Cdc,Elu,Spo,Dia,HEAT,DTT,COLD}\}} C_f(C, m)$, and $\mathcal{C}$ is the set of all possible 247 non-empty non-single combinations of data sets. That is, the second, third, and fourth columns show the number of combinations that give poorer, equal, and better performance—according to the $S(M, 2)$ measure—than the best of the 8 individual data sets (abbreviated as BI in the tables). The fifth, sixth, and seventh columns show the respective percentages with respect to the 247 total possible combinations.

Table 4.9: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 5 functions of yeast through SVM (EXH:Exhaustive study,BI:Best of individual data set).

| Function | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| HIST | 239 | 8 | 0 | 96.761 | 3.239 | 0.000 |
| PROT | 36 | 25 | 186 | 14.575 | 10.121 | 75.304 |
| RESP | 203 | 44 | 0 | 82.186 | 17.814 | 0.000 |
| RIBO | 2 | 1 | 244 | 0.810 | 0.405 | 98.785 |
| TCA | 199 | 10 | 38 | 80.567 | 4.049 | 15.385 |

We can see from these 4 tables that, out of the 4940 combinations through all four

Table 4.10: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 5 functions of yeast through NBay.

| Function | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| HIST | 236 | 4 | 7 | 95.547 | 1.619 | 2.834 |
| PROT | 221 | 1 | 25 | 89.474 | 0.405 | 10.121 |
| RESP | 246 | 1 | 0 | 99.595 | 0.405 | 0.000 |
| RIBO | 38 | 1 | 208 | 15.385 | 0.405 | 84.211 |
| TCA | 247 | 0 | 0 | 100.000 | 0.000 | 0.000 |

Table 4.11: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 5 functions of yeast through C4.5.

| Function | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| HIST | 236 | 4 | 7 | 95.547 | 1.619 | 2.834 |
| PROT | 168 | 5 | 74 | 68.016 | 2.024 | 29.960 |
| RESP | 135 | 111 | 1 | 54.656 | 44.939 | 0.405 |
| RIBO | 44 | 3 | 200 | 17.814 | 1.215 | 80.972 |
| TCA | 228 | 9 | 10 | 92.308 | 3.644 | 4.049 |

Table 4.12: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 5 functions of yeast through MLP.

| Function | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| HIST | 232 | 15 | 0 | 93.927 | 6.073 | 0.000 |
| PROT | 224 | 2 | 21 | 90.688 | 0.810 | 8.502 |
| RESP | 237 | 9 | 1 | 95.951 | 3.644 | 0.405 |
| RIBO | 18 | 0 | 229 | 7.287 | 0.000 | 92.713 |
| TCA | 227 | 5 | 15 | 91.903 | 2.024 | 6.073 |

algorithms on 5 functions by exhaustive search, 258 (=5%) of the possible combinations yield equal $S(M, 2)$ to best individual data sets, and 1266 (=26%) of the possible combinations yield higher $S(M, 2)$ than best individual data sets, and 3416 (=69%) of total combinations yield lesser $S(M, 2)$ than best individual data set. This means that, there is a decent 26% chance that using additional data—even when these are from experimental conditions (different sets of features on same set of genes)—can yield better classification models, at least for the purpose of predicting the 5 specific functions of yeast genes.

### 4.2.2 Use of Additional Micro-Environment Property on 3 Types of Protein Sites

In Subsection 4.1.2, we illustrated the accuracy of inferring if an atom has one of the 3 types of specific sites—CALCIUM, DISULFIDE, SERINE—based on a feature vector derived from one of 6 categories of micro-environment properties. It is reasonable to ask if one can perform better by using a feature vector derived by combining or merging two or more categories of properties on same site. That is, does the use of additional data (on same set of sites) improve classification accuracy on these 3 types of protein sites?

Recall that, there are 6 data sets corresponding to 6 categories of micro-environment properties—$D \in$ {Atom, Chemical, Residue, Sec-str, Others, Co-ord}. This yields 57 $(= 2^6 - 7)$ non-empty combinations that involved more than 1 of these 6 data sets. This is a sufficiently small number of combinations. So we are able to consider the performance by $S(M, 2)$ of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 3 types of protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE} using these 57 combinations of data sets, and compare the results with using the 6

data sets individually. The results from this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 4.13, 4.14, 4.15, and 4.16. In each of these 4 tables, the second column shows $|\{C_s(D,m) < l_s \mid D \in \mathcal{D}\}|$, the third column shows $|\{C_s(D,m) = l_s| \ D \in \mathcal{D}\}|$, and the fourth column shows $|\{C_s(D,m) > l_s| \ D \in \mathcal{D}\}|$, where $l_s = \max_{D \in \{\text{Atom,Chemical,Residue,Sec-str,Others,Co-ord}\}} C_s(D,m)$, and $\mathcal{D}$ is the set of all possible 57 non-empty non-single combinations of data sets. That is, the second, third, and fourth columns show the number of combinations that give poorer, equal, and better performance—according to the $S(M,2)$ measure—than the best of the 6 individual data sets (abbreviated as BI in the tables). The fifth, sixth, and seventh columns show the respective percentages with respect to the 57 total possible combinations.

Table 4.13: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 3 types of protein sites through SVM (EXH:Exhaustive study,BI:Best of individual data set).

| Prot-site | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| CALCIUM | 11 | 0 | 46 | 19.298 | 0.000 | 80.702 |
| SERINE | 11 | 1 | 45 | 19.298 | 1.754 | 78.947 |
| DISULFIDE | 6 | 0 | 51 | 10.526 | 0.000 | 89.474 |

Table 4.14: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 3 types of protein sites through NBay.

| Prot-site | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| CALCIUM | 20 | 2 | 35 | 35.088 | 3.509 | 61.404 |
| SERINE | 54 | 0 | 3 | 94.737 | 0.000 | 5.263 |
| DISULFIDE | 57 | 0 | 0 | 100.000 | 0.000 | 0.000 |

Table 4.15: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 3 types of protein sites through C4.5.

| Prot-site | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| CALCIUM | 51 | 2 | 4 | 89.474 | 3.509 | 7.018 |
| SERINE | 52 | 3 | 2 | 91.228 | 5.263 | 3.509 |
| DISULFIDE | 42 | 0 | 15 | 73.684 | 0.000 | 26.316 |

Table 4.16: Number and percentage for EXH<BI, EXH=BI, and EXH>BI on 3 types of protein sites through MLP.

| Prot-site | Les. ind. | Eq ind. | Grt. ind. | Les. ind. % | Eq ind.% | Grt. ind.% |
|---|---|---|---|---|---|---|
| CALCIUM | 53 | 0 | 4 | 92.982 | 0.000 | 7.018 |
| SERINE | 50 | 3 | 4 | 87.719 | 5.263 | 7.018 |
| DISULFIDE | 51 | 2 | 4 | 89.474 | 3.509 | 7.018 |

We can see from these 4 tables that, out of the 684 combinations through all four learning algorithms on 3 types of protein sites by exhaustive search, 213 (=31%) of the possible combination yield $S(M, 2)$ greater than the best individual data sets, and 13 (=2%) of the possible combination yield $S(M, 2)$ equals to the best individual data sets, and 458 (=67%) of the possible combinations yield lesser $S(M, 2)$ than best individual data set. This means that, there is a decent 31% chance that using additional data—even though these are from different categories of properties (on same set of sites)—can yield better classification models, at least for the purpose of predicting the 3 types of specific protein sites.

### 4.2.3 Use of Additional Microarray Data Sets on 26 Functions of Yeast Genes

In Subsection 4.1.3 we provided the accuracy of inferring if a gene has one of the 26 functions based on a feature vector derived from gene expression profile of that gene under one experimental condition. It is natural to speculate if one can perform better by using a feature vector derived by combining or merging gene expression profiles of that gene under two or more experimental conditions (different sets of features on same set of genes). That is, does the use of additional data (on same set of genes) improve classification accuracy for these 26 cellular functions?

Recall, that we have already mentioned for a larger number of data sets, it is not practical to do exhaustive search. For 57 data sets, there are millions of combinations ($= 2^{57} - 58$) that involve more than 1 of these data sets. This is a quite a large number of combinations.

In Chapter 5, in Section **5.1.2**, we develop a Greedy-Hill climbing algorithm to pick a good combination of data sets. We show in Section 5.5.2 comparison of Greedy-Hill to best individual data set.

## 4.3 Random Sampling and Incremental Strategies for Choosing Additional Data Sets

In previous sections, foregoing discussion shows that 26% of the possible combinations of data sets lead to better prediction accuracy than any of the individual data sets, and 5% of the possible combinations lead to equal prediction accuracy, on predicting 5 functions of yeast and 3 types of protein sites, respectively. However, this also means that on 5 functions of yeast problem, 69% of the combinations lead to worse

prediction accuracy than using the best of the individual data sets. In other words, if we pick a combination of data sets at randomly, we have a 69% chance of doing worse. In our example, there are only 247 possible combinations to consider and thus exhaustive testing is possible. This is no longer possible if we have many more data sets to consider. Therefore, we need a strategy to pick the better combinations of data sets.

### 4.3.1 5 Functions of Yeast Genes

One of the simplest strategy is that of sampling. For example, we can randomly sample a small percentage of the possible combinations, test the prediction accuracy of the sampled combinations, and use the combination that produces the best predication accuracy amongst the sampled combinations to be the prediction model. That is, we define a sampling-based classifier $C_f^{random}(\mathcal{C}, m)$ for the function $f$, machine learning method $m$, and set of possible combinations $\mathcal{C}$, such that $C_f^{random}(\mathcal{C}, m) = C_f(c^{random}, m)$ where $c^{random}$ is the best among 10 randomly sampled combinations from $\mathcal{C}$. The performance by $S(M, 2)$ for $C_f^{random}(\mathcal{C}, m)$ is then obtained. Since $C_f^{random}(\mathcal{C}, m)$ is dependent on the choice of the 10 sampled combinations, in order to properly evaluate the performance of this strategy, we repeat this process 100 times. Table 4.17 shows the results, where we report the percentage of times amongst the 100 repeats that $C_f^{random}(\mathcal{C}, m)$ achieves a $S(M, 2)$ measure that is equal to (Column 2) or better than (Column 3) the best of individual data sets.

We can see from the table that, for the function RIBO, the sampling-based classifier built on top of C4.5 (SVM, NBay, MLP, respectively) has a 80.66% (98.95%, 84.23%, 93.6%, respectively) chance of achieving better $S(M, 2)$ than the best of the C4.5 classifiers training from individual data sets. Thus for the function RIBO, the

Table 4.17: Percentage of 100 repeats of $C_f^{random}(\mathcal{C}, m)$ that is equal to or better than the best of individual data sets.

| | Eq. ind. % | | | | Grt. ind. % | | | |
|---|---|---|---|---|---|---|---|---|
| Function | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| HIST | 1.42 | 3.4 | 1.66 | 5.69 | 2.9 | 0 | 2.83 | 0 |
| PROT | 1.98 | 10.01 | 0.4 | 0.77 | 29.72 | 74.41 | 10.13 | 8.69 |
| RESP | 44.95 | 17.49 | 0.33 | 3.66 | 0 | 0 | 0 | 0.43 |
| RIBO | 1.33 | 0.33 | 0.35 | 0 | 80.66 | 98.95 | 84.23 | 93.6 |
| TCA | 3.7 | 3.94 | 0 | 2.05 | 3.59 | 14.96 | 0 | 5.72 |

sampling-based strategy works well. However, for functions HIST, PROT, RESP, and TCA, it is clear that the sampling-based strategy does not work very well.

Another simple strategy to pick better combinations of data sets is to incrementally add one data set at a time in a fixed but arbitrary order. To study this strategy, we perform a "stability" analysis. Let $C_1$, $C_2$, ..., be a chain of combinations of data sets so that $C_{i+1}$ comprises the data sets chosen in the combination $C_i$ and one additional data sets. Then we say that a learning method $m$ is stable for a function $f$ and chain $C_1$, $C_2$, ..., if the $S(M, 2)$ of $C_f(C_1, m)$, $C_f(C_2, m)$, ..., strictly increases. We generated 30 distinct chains from our 8 individual data sets, {Alp, Cdc, Elu, Spo, Dia, HEAT, DTT, COLD}. Then for each function $f$ and learning method $m$, we test if $m$ is stable for $f$ on these 30 chains. Table 4.18 gives results of our stability analysis on the 5 functions and 4 learning methods. In particular, for each $f$ and $m$, the table gives the percentage of chains amongst the 30 that $m$ is stable for $f$.

We can see from the table that for the function HIST, this "add one data set at a time in a fixed order" strategy has good stability. That is, for HIST, the performance by $S(M, 2)$ through all our learning methods strictly improves as we add one data set at a time in a fixed order. However, for the other functions, we do not see this

Table 4.18: Stability of the "add one data set at a time in a fixed order" strategy.

| Function | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|
| HIST | 76.67 | 90.00 | 73.33 | 83.33 |
| PROT | 53.33 | 70.00 | 63.33 | 43.33 |
| RESP | 33.33 | 56.67 | 20.00 | 30.00 |
| RIBO | 10.00 | 3.33 | 13.33 | 3.33 |
| TCA | 33.33 | 43.33 | 6.67 | 33.33 |

stability based on the 30 chains produced by the fixed order that we have chosen.

From above several analysis and illustrations, we come to a few conclusions. Exhaustive search through all possible combinations is obviously impractical if we have a large number of individual data sets. Random sampling in most cases does not improve performance. Incrementally adding one data set at a time in most cases does not improve performance, at least not when a fixed order is used. Therefore, a technique to efficiently search through the space of all possible combinations and picking a good one is needed. We will take up this challenge later in Chapter 5.

### 4.3.2 3 Types of Protein Sites

The earlier discussion shows that 31% of the possible combinations of data sets can lead to higher prediction accuracy than any of the individual data sets, and 2% of the possible combinations lead to equal prediction accuracy. This clearly indicates that 67% of the combination lead to worse prediction accuracy than using the best of the individual data sets. This cautions that if we pick a combination randomly, we have a 67% chance of doing worse. In our example, there are only 57 possible combinations to consider and thus exhaustive testing is possible. Exhaustive search is no longer possible if we have many more data sets to consider. Therefore, we need

a strategy to pick better combinations of data sets.

One of the simplest strategies is that of sampling. For example, we can randomly sample a small percentage of the possible combinations, test the prediction accuracy of the sampled combinations, and use the combination that produces the best prediction accuracy amongst the sampled combinations to be the prediction model. That is, we define a sampling-based classifier $C_s^{random}(\mathcal{D}, m)$ for the protein site $s$, machine learning method $m$, and set of possible combinations $\mathcal{D}$, such that $C_s^{random}(\mathcal{D}, m) = C_s(d^{random}, m)$ where $d^{random}$ is the best among 10 randomly sampled combinations from $\mathcal{D}$. The performance by $S(M, 2)$ for $C_s^{random}(\mathcal{D}, m)$ is then obtained. Since $C_s^{random}(\mathcal{D}, m)$ is dependent on the choice of the 10 sampled combinations, in order to properly evaluate the performance of this strategy, we repeat this process 100 times. Table 4.19 shows the results, where we report the percentage of times amongst the 100 repeats that $C_s^{random}(\mathcal{D}, m)$ achieves a $S(M, 2)$ measure that is equal to (Column 2) or better than (Column 3) the best of individual data sets.

Table 4.19: Percentage of 100 repeats of $C_s^{random}(\mathcal{D}, m)$ that is equal to or better than the best of individual data sets.

| | Eq. ind.% | | | | Grt. ind.% | | | |
|---|---|---|---|---|---|---|---|---|
| Prot-site | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| CALCIUM | 3.32 | 0 | 3.45 | 0 | 6.35 | 82.25 | 63.02 | 6.91 |
| SERINE | 5.58 | 1.72 | 0 | 5.11 | 2.48 | 78.45 | 5.44 | 6.36 |
| DISULFIDE | 0 | 0 | 0 | 2.66 | 24.9 | 89.02 | 0 | 6.42 |

Table 4.19 shows clearly that sampling-based strategy does not work very well in general for protein sites CALCIUM, SERINE, and DISULFIDE.

Another simple strategy to choose better combinations of data sets is to incrementally "add one data set at a time in a fixed but arbitrary order". To study this

strategy, we do a "stability" analysis. Let, $D_1$, $D_2$, ..., be a chain of combinations of data sets so that $D_{i+1}$ comprises the data sets chosen in the combination $D_i$ and one additional data set. Now, we say that a learning method $m$ is stable for a protein site $s$ and chain $D_1$, $D_2$, ..., if the $S(M, 2)$ of $C_s(D_1, m)$, $C_s(D_2, m)$, ..., strictly increases. We generated 30 distinct chains from our 6 individual data sets, $D \in \{$Atom, Chemical, Residue, Sec-str, Others, Co-ord$\}$. Then for each protein site $s$ and learning method $m$, we test if $m$ is stable for $s$ on these 30 chains. Table 4.20 gives results of our "stability" analysis on the 3 types of protein sites and 4 learning methods. In particular, for each $s$ and $m$, the table gives the percentage of chains amongst the 30 that $m$ is stable for $s$.

Table 4.20: "Stability" of the "add one data set at at time in a fixed order" strategy.

| Prot-site | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|
| CALCIUM | 76.92 | 69.23 | 23.08 | 76.92 |
| SERINE | 76.92 | 61.54 | 92.31 | 92.31 |
| DISULFIDE | 69.23 | 61.54 | 69.23 | 61.54 |

We can see from Table 4.20 that for protein site SERINE, this "add one data set at a time in a fixed order" strategy has good stability. That is, for SERINE, the performance by $S(M, 2)$ through all our learning algorithms strictly improves as we add one data set at a time in a fixed order. However, for protein sites CALCIUM and DISULFIDE, we do not see this stability based on the 30 chains produced by the fixed order that we have chosen.

From the several analysis above on protein sites, we come to a few conclusions. Exhaustive search through all possible combinations is obviously impractical when we have a large number of data sets. Random sampling may not always improve

performance. Incrementally adding "one data set at a time in fixed order" may not always improve performance. A better methodology to search through possible combinations and pick a best one is needed. We will address this issue in Chapter 5.

## 4.4 Using ALL Data in Modeling

In Section 4.2, we demonstrated that the use of additional or combination of data sets—even when data sets are derived from experiments (different sets of features on same set of genes or sites)—can increase prediction accuracy of classification models on a variety of bioinformatics problems. This seems to suggest that using all available data sets may be a simple way to improve prediction accuracy. However, in this section, we show how using all available data sets does not give the best improved prediction accuracy and often gives a worse accuracy than using the best individual data sets.

### 4.4.1 Use of ALL 6 Microarray Data Sets on 5 Functions of Yeast Genes

The focus of this section is to analyse prediction accuracy on classification models using all available data sets, and show its demerits. We now combine all those 6 sets of gene expression experiments—from the paper of Eisen *et al.*—{Alp, cdc, Elu, ccc, Spo, Dia}—based on the 6 experimental conditions (different sets of features on same set of genes), as depicted in Table 3.1, into one single data set and name it as $ALL$.

We show below in Table 4.21 the results of the experiments just described, using SVM and MLP as the learning methods, and $ALL$ as the data set. We can think of this table as the performance by $S(M, 2)$ of SVM and MLP for predicting 5 specific cellular

functions of yeast genes using the $ALL$ data set. The rows are the 5 functions—HIST (histones), PROT (proteosomes), RESP (respiration), RIBO (ribosomes), and TCA (TCA cycle). The columns 2 and 3 show the performance by $S(M, 2)$ of $ALL$ by SVM and MLP for 5 functions of yeast. The SVM here is the support vector machine implementation from the GIST package and uses RBF of degree 3, as in Brown *et al.* Here MLP is the multilayer perceptron of WEKA implementation at its default settings. Our annotations are based on MIPS Catalogue (Version 1.3) of 25th June 2003.

Table 4.21: Performance by $S(M, 2)$ on 5 functions of yeast based on $ALL$ data sets through SVM and MLP.

| Function | SVM | MLP |
|---|---|---|
| HIST | 17 | 17 |
| PROT | 25 | 30 |
| RESP | -103 | -12 |
| RIBO | 217 | 209 |
| TCA | -9 | -2 |

Earlier, we gave the accuracy of inferring if a gene has one of the 5 specific functions—HIST, PROT, RESP, RIBO, or TCA—based on a feature vector derived from gene expression profile of that gene under the combined $ALL$ data set. Since we have shown in the previous subsection that adding additional data sets (on same set of genes) can improve prediction performance by $S(M, 2)$, it is natural to ask whether using all available data sets can lead to the best performance by $S(M, 2)$. Now, we generate all possible combinations of 2 or more data sets, but exclude the combination that uses all available data sets, and compare their prediction performance to that of using all available data sets.

Recall that in Brown *et al.*, with genes annotated as per MIPS Catalogue (Version 1.3) of 25th June 2003, there are 8 data sets corresponding to 8 experimental conditions (different sets of features on same set of genes). This yields 246 ($= 2^8 - 10$) non-empty combinations that involve more than 1 of these 8 data sets but do not involve all 8 data sets. We have a sufficiently small number of combinations here. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in \{$C4.5, SVM, NBay, and MLP$\}$ for predicting the 5 specific functions $f \in \{$HIST, PROT, RESP, RIBO, TCA$\}$ using these 246 combinations of data sets, and compare the results with using the $ALL$ data set. Results through this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 4.22, 4.24, 4.23, and 4.25. In each of these 4 tables, the second column shows $|\{C_f(C, m) < k_f |\ C \in \mathcal{C}\}|$, the third column shows $|\{C_f(C, m) = k_f |\ C \in \mathcal{C}\}|$, and the fourth column shows $|\{C_f(C, m) > k_f |\ C \in \mathcal{C}\}|$, where $k_f = C_f(ALL, m)$, $\mathcal{C}$ is the set of all possible 246 non-empty non-single combinations of data sets, and $\mathcal{C}$ is the collection of all the combinations that involve at least 2 data sets but not all 8 data sets. That is, the second, third, and fourth columns show the number of combinations that give poorer, equal, and better performance—according to the $S(M, 2)$ measure—than $ALL$ data sets (abbreviated as ALL in the tables). The fifth, sixth, and seventh columns show the respective percentages with respect to the 246 total possible combinations.

From the 4 tables we can see that out of the 4920 exhaustive comparisons over SVM, C4.5, NBay, and MLP on the 5 specific cellular functions, 81 (=2%) of the possible combinations of data sets yield an accuracy equal to the $ALL$ data set, 1258 (=26%) of the possible combinations of data sets yield better accuracy than the $ALL$ data set, and 3581 (=72%) of total combinations yield lesser accuracy than $ALL$ data

Table 4.22: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 5 functions of yeast through SVM (EXH: Exhaustive study, ALL: All data sets).

| Function | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| HIST | 236 | 2 | 8 | 95.93 | 0.81 | 3.25 |
| PROT | 238 | 1 | 7 | 96.75 | 0.41 | 2.85 |
| RESP | 15 | 1 | 230 | 6.10 | 0.41 | 93.50 |
| RIBO | 174 | 10 | 62 | 70.73 | 4.07 | 25.20 |
| TCA | 222 | 1 | 23 | 90.24 | 0.41 | 9.35 |

Table 4.23: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 5 functions of yeast through NBay.

| Function | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| HIST | 246 | 0 | 0 | 100.00 | 0.00 | 0.00 |
| PROT | 244 | 0 | 2 | 99.19 | 0.00 | 0.81 |
| RESP | 84 | 3 | 159 | 34.15 | 1.22 | 64.63 |
| RIBO | 170 | 7 | 69 | 69.11 | 2.85 | 28.05 |
| TCA | 238 | 0 | 8 | 96.75 | 0.00 | 3.25 |

Table 4.24: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 5 functions of yeast through C4.5.

| Function | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| HIST | 239 | 0 | 7 | 97.15 | 0.00 | 2.85 |
| PROT | 238 | 4 | 4 | 96.75 | 1.63 | 1.63 |
| RESP | 62 | 14 | 170 | 25.20 | 5.69 | 69.11 |
| RIBO | 155 | 10 | 81 | 63.01 | 4.07 | 32.93 |
| TCA | 246 | 0 | 0 | 100.00 | 0.00 | 0.00 |

Table 4.25: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 5 functions of yeast through MLP.

| Function | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| HIST | 231 | 15 | 0 | 93.90 | 6.10 | 0.00 |
| PROT | 233 | 3 | 10 | 94.72 | 1.22 | 4.07 |
| RESP | 2 | 3 | 241 | 0.81 | 1.22 | 97.97 |
| RIBO | 76 | 2 | 168 | 30.89 | 0.81 | 68.29 |
| TCA | 232 | 5 | 9 | 94.31 | 2.03 | 3.66 |

set.

### 4.4.2 Use of ALL Micro-environment Properties on 3 Types of Protein Sites

The focus of this section is to analyse prediction accuracy on protein sites using all available data sets, and show its demerits. Now we combine all the 6 sets of micro-environment properties into one single data set and named it as $ALL$. We show below in Table 4.26 the results, using different learning methods, and the $ALL$ data set. We can think of this table as the performance by $S(M, 2)$ for predicting 3 types of protein sites using $ALL$ data set. The rows are the 3 types of protein sites—CALCIUM (calcium binding sites), DISULFIDE (disulfide bridges) and SERINE (serine protease active sites). The second through the fifth columns are the performance by different learning algorithms through WEKA package on $ALL$ data set.

Earlier, we gave the accuracy of inferring if an atom has one of the 3 types of protein sites—CALCIUM, DISULFIDE, SERINE—based on a feature vector derived from micro-environment properties of that atom under the $ALL$ data set. Since we have shown in the previous subsection that adding additional data sets (on same

Table 4.26: Performance by $S(M, 2)$ on 3 types of protein sites based on $ALL$ data sets through algorithms.

| Sites | C4.5 | NBay | SVM | MLP |
|---|---|---|---|---|
| CALCIUM | 181 | 154 | 153 | 167 |
| SERINE | 66 | 45 | 66 | 68 |
| DISULFIDE | 83 | 65 | 60 | 71 |

set of sites) can improve prediction performance by $S(M, 2)$, it is natural to ask whether using all available data sets can lead to the best performance. So we now generate all possible combinations of 2 or more data sets, but exclude the combination that uses all available data sets, and compare their prediction performance to that of using all available data sets. We recall that for annotated atoms there are 6 data sets corresponding to 6 micro-environment properties—$D \in$ {Atom, Chemical, Residue, Sec-str, Others, Co-ord}. This yields 56 ($= 2^6 - 8$) non-empty combinations that involve more than 1 of these 6 data sets but do not involve all 6 of them. This is a sufficiently small number of combinations. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 3 types of protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE} using these 56 combinations of data sets, and compare the results with using $ALL$ data set. Results through this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 4.27, 4.28, 4.29, and 4.30. In each of these 4 tables, the second column shows $|\{C_s(C, m) < l_s \mid C \in \mathcal{C}\}|$, the third column shows $|\{C_s(C, m) = l_s| \ C \in \mathcal{C}\}|$, and the fourth column shows $|\{C_s(C, m) > l_s| \ C \in \mathcal{C}\}|$, where $l_s = C_s(ALL, m)$, and $\mathcal{C}$ is the set of all possible 56 non-empty non-single combinations of data sets that do not involve all 6 data sets, and $ALL$ is the combined data set. That is, the

second, third, and fourth columns show the number of combinations that give poorer, equal, and better performance—according to the $S(M, 2)$ measure—than $ALL$ data sets (abbreviated as ALL in the tables). The fifth, sixth, and seventh columns show the respective percentages with respect to the 56 total possible combinations.

Table 4.27: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 3 types of protein sites through SVM (EXH: Exhaustive study, ALL: All data sets).

| Prot-site | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| CALCIUM | 35 | 2 | 19 | 62.500 | 3.571 | 33.929 |
| SERINE | 51 | 1 | 4 | 91.071 | 1.786 | 7.143 |
| DISULFIDE | 56 | 0 | 0 | 100.000 | 0.000 | 0.000 |

Table 4.28: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 3 types of protein sites through NBay.

| Prot-site | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| CALCIUM | 53 | 0 | 3 | 94.643 | 0.000 | 5.357 |
| SERINE | 45 | 1 | 10 | 80.357 | 1.786 | 17.857 |
| DISULFIDE | 49 | 2 | 5 | 87.500 | 3.571 | 8.929 |

We can see from these 4 tables that, out of the 672 exhaustive comparisons over SVM, C4.5, NBay, and MLP on the 3 types of protein sites, 12 (=2%) of the possible combination of data sets yield an accuracy equal to the $ALL$ data set, 71 (=11%) of the possible combination of data sets yield better accuracy than $ALL$ data set, and 589 (=87%) of total combinations yield lesser accuracy than $ALL$ data set.

Table 4.29: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 3 types of protein sites through C4.5.

| Prot-site | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| CALCIUM | 52 | 2 | 2 | 92.857 | 3.571 | 3.571 |
| SERINE | 54 | 1 | 1 | 96.429 | 1.786 | 1.786 |
| DISULFIDE | 50 | 2 | 4 | 89.286 | 3.571 | 7.143 |

Table 4.30: Number and percentage for EXH<ALL, EXH=ALL, and EXH>ALL on 3 types of protein sites through MLP.

| Prot-site | Les. ALL | Eq ALL | Grt. ALL | Les. ALL% | Eq ALL% | Grt. ALL% |
|---|---|---|---|---|---|---|
| CALCIUM | 50 | 0 | 6 | 89.286 | 0.000 | 10.714 |
| SERINE | 38 | 1 | 17 | 67.857 | 1.786 | 30.357 |
| DISULFIDE | 56 | 0 | 0 | 100.000 | 0.000 | 0.000 |

### 4.4.3 Use of ALL 57 Microarray Data Sets on 26 Functions of Yeast Genes

In this section, we discuss classification studies in which many more functions of yeast are involved than the earlier studies with 5 functions (as in Section 4.1.1).

The focus of this section is to analyse prediction accuracy on classification models using all available data set and show its demerits. We now combine all those 57 data sets of gene expression experiments—{Alp1, Alp2, Alp3, Alp4, Ace, Des1, Des2, Des3, Des4, Des5, Des6, Des7, Haa, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15, Hea16, Hea17, Hea18, Hea19, Hea20, Dby1, Dby2, Dby3, Dby4, Dby5, Dby6, Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8, Fch1, Fch2, Met, Hyd, Iro, Aft, Fit, Pho, Snf, Spo}—based on the 57 experimental conditions (different sets of features on same set of genes) (as

detailed in Table 3.6), into one single data set and name it as $ALL$.

We show below in Table 4.31 the results of $C_f(ALL, m)$, for 26 functions, where $m \in$ {C4.5, SVM, NBay, MLP}, for 26 functions $f \in$ {Aam, Nsm, Nuc, Pho, Ccm, Lim, Vit, Tca, Res, Fer, Mer, Ecr, Dna, Cyc, Tcp, Rsn, Rpr, Rmo, Pro, Rib, Tra, Trc, Ami, Pft, Pfs, Ptt, Prm, Apc, Deg, Tcs, Tfc, Trt, Int, Rdv, Str, Dtx, Hom, Csr, Tvp, Gro, Dea, Wal, Cyt, Nuc, Mit, Fun} annotated as per MIPS Catalogue (Version 2.0) of 19th March 2004, and for the data set $ALL$ derived by combining or merging 57 (different sets of features on same set of genes) data sets. In this table, we show the $S(M, 2)$ score based on the performance of $m$ by $S(M, 2)$ on the $ALL$ data set; that is, for a function $f$, $S(M, 2) = C_f(ALL, m)$.

Table 4.31: Performance by $S(M, 2)$ on 26 functions of yeast based on $ALL$ data sets through algorithms.

| Function | Code | Genes | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | 20 | 16 | 3 | 7 |
| 11.04 | Rpr | 161 | 14 | 0 | 8 | 18 |
| 10.03 | Cyc | 149 | 7 | 11 | 2 | 9 |
| 20.09 | Trt | 145 | 0 | 2 | 4 | 2 |
| 12.01 | Rib | 138 | 221 | 239 | 208 | 249 |
| 1.01 | Aam | 103 | 65 | 59 | 11 | 74 |
| 1.06 | Lim | 99 | 18 | 0 | 0 | 20 |
| 10.01 | Dna | 99 | 17 | 3 | 4 | 20 |
| 1.05 | Ccm | 82 | 2 | 2 | 0 | 4 |
| 1.03 | Nuc | 81 | 13 | 14 | 0 | 22 |
| 14.13 | Deg | 77 | 32 | 0 | 7 | 38 |
| 32.01 | Str | 58 | 3 | 7 | 1 | 8 |
| 1.07 | Vit | 54 | 0 | 0 | 0 | 0 |
| 14.07 | Prm | 48 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | 0 | 0 | 0 | 13 |
| 12.04 | Tra | 42 | 0 | 0 | 1 | 8 |
| 11 | Tcp | 39 | 0 | 0 | 0 | 1 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | 15 | 15 | 0 | 16 |
| 20.03 | Tfc | 32 | 0 | 0 | 0 | 0 |
| 42.01 | Wal | 32 | 0 | 0 | 0 | 0 |
| 12.10 | Ami | 31 | 13 | 5 | 0 | 21 |
| 43.01 | Fun | 31 | 0 | 0 | 0 | 1 |
| 2.13 | Res | 29 | 12 | 11 | 4 | 25 |
| 14.01 | Pfs | 29 | 0 | 0 | 0 | 4 |
| 32.07 | Dtx | 27 | 0 | 8 | 0 | 4 |

Table 4.32: Number and percentage over 26 functions of yeast for BI>ALL, BI=ALL, and BI<ALL through algorithms.

| | Grt. *ALL* | | | | Eq. *ALL* | | | | Les. *ALL* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| 26 Functions | 25 | 22 | 25 | 18 | 0 | 0 | 0 | 2 | 1 | 4 | 1 | 6 |
| % Functions | 96 | 85 | 96 | 69 | 0 | 0 | 0 | 8 | 4 | 15 | 4 | 23 |

Table 4.32 shows that 87%, 2%, and 11% of functions over all 4 algorithms $BI$ is greater, equal and lesser than $ALL$.

## 4.5 Using Selected Features from Conventional Feature Selection Methods

In Section 4.2, we demonstrated that the use of additional or multiple data sets—even when the data sets are derived from experiments (different sets of features on same set of genes or sites)—can increase prediction accuracy of classification models on a variety of bioinformatics problems. In Section 4.4, we cautioned that using all available data sets does not give the best improved prediction accuracy, and often gives a worse accuracy than using the best individual data sets or multiple data sets. The next choice generally used in classification problems by many researchers is to apply feature selection methods and build a model from the selected features. We illustrate, in this section, how prediction accuracy can be improved by using conventional feature selection methods compared to using the best individual data sets or all available data sets. However, we also show that conventional feature selection methods do not achieve the best prediction accuracy often enough than using a combination of whole individual data sets.

### 4.5.1 Use of Selected Features on 5 Functions of Yeast Genes

In Subsection 2.1.1 we briefly describe experiments of Brown *et al.* [5] and Mateos *et al.* [34], who attempted to infer 5 specific functions of yeast from 6 gene expression data sets. In this subsection we compare the performance by $S(M, 2)$ using selected features from conventional feature selection methods to that of using the best individual data sets and to that of using all available data sets. We also show how this conventional feature selection approach does not lead to the best classification accuracy often enough, by comparing it with the accuracy obtained by an exhaustive search through all possible combinations of whole data sets.

The focus of this section is to analyse prediction accuracy on classification models using features selected by conventional feature selection methods—CFS (Correlation-based feature selection), Chi (Chi-squared feature selection), Info (Information-gain feature selection), Fisher, and T-test (only on 5 functions of yeast)—and show their merits and demerits. In the Subsection 4.4.1, we derive the $ALL$ data set by combining or merging all 6 sets of gene expression experiments from the paper of Eisen *et al.*—{Alp, cdc, Elu, ccc, Spo, Dia}—based on the 6 experimental conditions (different sets of features on same set of genes), as depicted in Table 3.1. Now to build a classifier $C_f(C, FS + m)$ based on features selected by a feature selection method $FS$ using a machine learning method $m$ on a data set $C$, the feature selection method $FS$ is applied on the training portion of the data to obtained a reduced data set, the machine learning method $m$ is then applied to the reduced data set to obtain a classification model which is then applied to the testing portion of the data.

Now we apply conventional feature selection methods $FS \in$ {Fisher, T-test} (Note: GIST doest not support CFS, Chi, and Info) on $ALL$ data set, and build

classification models $C_f(ALL, FS + m)$. After a number of initial experiments, we have determined that using the top 8 features selected by the *Fisher* and *T-test* methods in the feature selection step would produce the best performance. We show below in Table 4.33 the results of using these top 8 features, with SVM as the learning method. We can think of this table as the performance by $S(M, 2)$ of SVM for predicting 5 specific cellular functions of yeast genes using the top 8 features selected by the *Fisher* and *T-test* methods. The rows are the 5 functions—HIST (histones), PROT (proteosomes), RESP (respiration), RIBO (ribosomes), and TCA (TCA cycle). Columns 2 and 3 show the $S(FS+SVM, 2)$ performance of $FS$ by SVM for 5 functions of yeast. The SVM here is the support vector machine implementation from the GIST package (Release 2.0.5, April 30, 2003) that uses RBF of degree 3 as in Brown *et al.* Our annotations are based on MIPS catalogue (Version 1.3) of 25th June 2003.

Table 4.33: Performance by $S(M, 2)$ on 5 specific functions of yeast based on selected features through Fisher and T-test through SVM.

| Function | T-test | Fisher |
|---|---|---|
| HIST | 0 | 0 |
| PROT | 2 | -5 |
| RESP | -922 | -696 |
| RIBO | -98 | -105 |
| TCA | -37 | -36 |

Earlier, we gave the accuracy of inferring if a gene has one of the 5 specific functions—HIST, PROT, RESP, RIBO, or TCA—based on a feature vector derived from the gene expression profile of that gene under the $FS$-selected data set. Now, we compare $FS$ results with that of using the best individual data sets and all available data sets.

In Brown *et al.*, with genes annotated as per MIPS Catalogue (Version 1.3) of 25th June 2003, there are 8 data sets corresponding to 8 experimental conditions (different sets of features on same set of genes)—$C \in$ {Alp, Cdc, Elu, Spo, Dia, HEAT, DTT, COLD}. We derive the best individual performance for a function $f$ as $S(M,2) = \max_{C \in \{\text{Alp,Cdc,Elu,Spo,Dia,HEAT,DTT,COLD}\}} C_f(C,m)$ among 8 data sets. We also got the performance for a function $f$, $S(FS{+}m,2) = C_f(ALL, FS{+}m)$, for $FS \in$ {CFS, Chi, Info} (Note: WEKA does not support Fisher, T-test) for learning methods, $m \in$ {C4.5, SVM, NBay, MLP}, and for a function $f \in$ {HIST, PROT, RESP, RIBO, TCA}. Our annotations are based on MIPS Catalogue (Version 1.3) of 25th June 2003. Table 4.34 shows the number of functions $f$ that achieve a $S(M,2)$ measure, based on the best individual data sets, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(FS+m,2)$ for the combination of feature selection method $FS$ and machine learning method $m$. Here, we use implementations from the WEKA package [65] with default settings. It is clear from the table that for most protein functions, the feature selection methods give a higher performance than using the best of individual data sets.

Table 4.34: Number and percentage for BI>FS, BI=FS, and BI<FS on 5 functions of yeast through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 0 | 1 | 2 | 0 | 1 | 1 | 0 | 2 | 4 | 3 | 3 | 3 |
| % Functions | 0 | 20 | 40 | 0 | 20 | 20 | 0 | 40 | 80 | 60 | 60 | 60 |

We calculate the performance by $S(M,2)$ for a function $f$ from $C_f(ALL,m)$, where $ALL$ is a data set derived by combining or merging 8 data sets—Alp, Cdc, Elu, Spo,

Dia, HEAT, DTT, and COLD based on same genes. We also get the performance $S(FS + m, 2)$ for a function $f$ from $C_f(ALL, FS + m)$, where $FS \in$ {CFS, Chi, Info} (Note: WEKA does not support Fisher, T-test) for various learning methods, $m \in$ {C4.5, SVM, NBay, MLP} and a function $f \in$ {HIST, PROT, RESP, RIBO, TCA}. Our annotations are based on MIPS Catalogue (Version 1.3) of 25th June 2003. Table 4.35 shows number of functions $f$ that achieve a performance by $S(M, 2)$ measure from $C_f(ALL, m)$ based on using all available data sets, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(FS + m, 2)$ for the combination of feature selection method $FS$ and machine learning method $m$. Here we use implementation of WEKA package [65] with default settings. It is clear from the table that for most protein functions, the use of a feature selection method gives a higher performance than using all available data sets.

Table 4.35: Number and percentage for ALL>FS, ALL=FS, and ALL<FS on 5 functions of yeast through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 0 | 2 | 1 | 0 | 1 | 0 | 0 | 1 | 4 | 3 | 4 | 4 |
| % Functions | 0 | 40 | 20 | 0 | 20 | 0 | 0 | 20 | 80 | 60 | 80 | 80 |

Earlier, we showed the accuracy of inferring if a gene has one of the 5 specific functions—HIST, PROT, RESP, RIBO, TCA—based on a feature vector derived from gene expression profile of that gene under the $FS$-selected data. Since we have shown in the previous subsection that using multiple data sets can improve prediction performance, it is natural to compare the best combination of multiple data sets to that of $FS$-selected data here. In this subsection, we generate all combinations of 2 or

more data sets, and compare the best prediction performance from exhaustive search through these combinations to the best performance from feature selection methods.

We have 8 data sets from Brown *et al.*, with genes annotated as per MIPS Catalogue (Version 1.3) of 25th June 2003, corresponding to 8 experimental conditions (different sets of features on same set of genes). This yields 255 ($= 2^8 - 1$) non-empty sets that involve the 8 data sets. We have a sufficiently small number of combinations of data sets here. This enables us to consider the performance of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 5 specific functions $f \in$ {HIST, PROT, RESP, RIBO, TCA} using these 255 sets of data sets, and compare the results with using $FS$-selected data. We also obtain the performance $S(FS + m, 2)$ for a function $f$ from $C_f(ALL, FS + m)$, where $FS \in$ {CFS, Chi, Info} (Note: WEKA does not support Fisher, T-test), for various learning methods $m \in$ {C4.5, SVM, NBay, MLP} and a function $f \in$ {HIST, PROT, RESP, RIBO, TCA}. We take the performance by $S(M, 2)$ of best exhaustive search on each function $f$ and compare it to the best performance on the same function from feature selection methods $S(FS + m, 2)$, where $FS \in$ {CFS, Chi, Info} and $m \in \{C4.5, SVM, NBay, MLP\}$.

Table 4.36 shows the number of functions $f$ that achieve a $S(M, 2)$ measure, based on the best combination of multiple whole data sets through an exhaustive search, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), than the performance $S(FS + m, 2)$ for the combination of feature selection method $FS \in$ {CFS, Chi, Info} and machine learning method $m \in \{C4.5, SVM, NBay, MLP\}$. Here we use implementation of WEKA package [65] with default settings. It is clear from the table that for a majority of protein functions, the best combination of multiple whole data sets achieves a better

performance than the best performance from conventional feature selection methods.

Table 4.36: Number and percentage for EXH>FS, EXH=FS, and EXH<FS on 5 functions of yeast through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 3 | 3 | 4 | 4 | 1 | 2 | 0 | 1 | 1 | 0 | 1 | 0 |
| % Functions | 60 | 60 | 80 | 80 | 20 | 40 | 0 | 20 | 20 | 0 | 20 | 0 |

### 4.5.2 Use of Selected Properties on 3 Types of Protein Sites

In Subsection 2.2.1, we recall briefly studies on micro-environment properties by Bagley *et al.* [57], calcium binding sites by Wei *et al.* [31], serine active sites by Bagley *et al.* [58], ATP binding sites and disulfide bonding sites by Wei *et al.* [29]. In this subsection we compare the performance by $S(M, 2)$ of using selected features from conventional feature selection methods to that of using the best individual data sets, and to that of using all available data sets. We also show how this conventional feature selection approach does not lead to the best classification accuracy often enough, by comparing it with the accuracy obtained by an exhaustive search through all possible combinations of whole micro-environment properties.

The focus of this section is to analyse prediction accuracy on protein sites using features selected by conventional feature selection methods—CFS (Correlation-based feature selection), Chi (Chi-squared feature selection), and Info (Information-gain feature selection)—and show their merits and demerits. In the previous subsection, we derived the $ALL$ data set by combining or merging 6 sets of micro-environment properties—$D \in$ {Atom, Chemical, Residue, Sec-str, Others, Co-ord} based on same site. Now we build a classifier $C_s(D, FS + m)$ based on features selected by a feature

selection method $FS$ and a machine learning method $m$ on a data set $D$. Here, the feature selection method $FS$ is applied on the training portion of the data to obtained a reduced data set, the machine learning method $m$ is then applied to the reduced data set to obtain a classification model which is applied to the testing portion of the data.

Now we apply conventional feature selection methods $FS \in$ {CFS, Chi, Info} (Note: WEKA does not support Fisher, T-test) on the $ALL$ data set and build classification models $C_s(ALL, FS{+}m)$. We show below in Table 4.37 the performance $S(FS + m, 2)$ of the experiments just described, using different learning methods on the $FS$-selected data sets. We can think of this table as the best performance by $S(M, 2)$ for predicting 3 types of protein sites using $FS$-selected data. The rows are the 3 types of protein sites—CALCIUM (calcium binding sites), SERINE (serine protease active sites), and DISULFIDE (disulfide bridges). The second through the fifth columns are the best performance by $S(M, 2)$ (among different feature selection methods) through WEKA package with default settings, given in the format "feature selection method:performance $S(FS + m, 2)$".

Table 4.37: Best performance by $S(M, 2)$ on 3 types of protein sites out of selected features through feature selection methods and algorithms.

| Prot-site | C4.5 | NBay | SVM | MLP |
|---|---|---|---|---|
| `CALCIUM` | CFS:184 | Chi:151 | Chi:165 | CFS :182 |
| `SERINE` | CFS: 83 | CFS: 78 | Chi: 65 | CFS : 85 |
| `DISULFIDE` | CFS: 67 | Chi: 44 | Chi: 65 | Info: 67 |

Earlier, we showed the accuracy of inferring if an atom has one of the 3 types of protein sites—CALCIUM, SERINE, and DISULFIDE—based on a feature vector

derived from micro-environment profile of that atom under $FS$-selected data. Now, we compare $FS$ results with that of using the best individual data sets, and with that of using all available data sets.

We have 6 data sets corresponding to 6 micro-environment properties—$D \in$ {Atom, Chemical, Residue, Sec-str, Others, Co-ord}. We get the performance for a site $s$ as $S(M,2) = \max_{D\in\{\text{Atom,Chemical,Residue,Sec-str,Others,Co-ord}\}} C_s(D,m)$, by using the best individual data sets and various learning methods $m \in$ {C4.5, SVM, NBay, MLP}, and for sites $s \in$ {CALCIUM, DISULFIDE, SERINE}. We also get the performance for a site $s$, $S(FS+m,2) = C_s(ALL, FS+m)$, for $FS \in$ {CFS, Chi, Info}, for various learning methods $m \in$ {C4.5, SVM, NBay, MLP}, for sites $s \in$ {CALCIUM, DISULFIDE, SERINE}. Table 4.38 shows the number of protein sites $s$ that achieve a $S(M,2)$ measure, based on the best set of micro-environment properties, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(FS+m,2)$ for the combination of feature selection method $FS$ and machine learning method $m$. Here we use the implementations in the WEKA package [65] with default settings. It is clear from the table that for all protein sites, the feature selection methods give higher performance by $S(M,2)$ than using the best individual sets of micro-environment properties.

Table 4.38: Number and percentage for BI>FS, BI=FS, and BI<FS on 3 types of protein sites through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 3 | 3 | 2 | 3 |
| % Sites | 0 | 0 | 33 | 0 | 0 | 0 | 0 | 0 | 100 | 100 | 67 | 100 |

We calculate the performance for a protein site $s$, $S(M,2) = C_s(ALL, m)$, where $ALL$ is a data set derived by combining or merging 6 data sets—Atom, Chemical, Residue, Sec-str, Others, and Co-ord based on same gene. We also got performance for a site $s$, $S(FS+m,2) = C_s(ALL, FS+m)$, where $FS \in$ {CFS, Chi, Info} for various learning methods $m \in$ {C4.5, SVM, NBay, MLP} and sites $s \in$ {CALCIUM, DISULFIDE, SERINE}. Table 4.39 shows the number of sites $s$ that achieve a $S(M,2)$ measure, based on using all available micro-environment properties, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(FS+m,2)$ for the combination of feature selection method $FS$ and machine learning method $m$. Here we use the implementations in the WEKA package [65] with default settings. It is clear from the table that for a majority of protein sites, conventional feature selection methods give higher performance by $S(M,2)$ than using all available micro-environment properties.

Table 4.39: Number and percentage for ALL>FS, ALL=FS, and ALL<FS on 3 types of protein sites through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 1 | 1 | 2 | 1 | 0 | 0 | 0 | 0 | 2 | 2 | 1 | 2 |
| % Sites | 33 | 33 | 67 | 33 | 0 | 0 | 0 | 0 | 67 | 67 | 33 | 67 |

Earlier, we showed the accuracy of inferring if an atom has one of the 3 types of protein sites—CALCIUM, SERINE, and DISULFIDE—based on a feature vector derived from the profile of that site under $FS$-selected data. Since we have shown in Subsection 4.2.2 that using multiple data sets can improve prediction performance, it is natural to compare the best combination of multiple data sets to that of $FS$-selected data here. Here, we generate all combinations of 2 or more data sets and

compare the best prediction performance from exhaustive search to that of the best performance from feature selection methods.

We have 6 data sets corresponding to 6 micro-environment properties—$D \in$ {Atom, Chemical, Residue, Sec-str, Others, Co-ord}. Now, this yields 63 ($= 2^6 - 1$) non-empty sets. This is a sufficiently small number of sets. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 3 types of protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE} using these 63 sets of sets, and compare the best performance on the same protein site through feature selection methods. We get the performance for a protein site $s$, $S(FS + m, 2) = C_s(ALL, FS + m)$, where $FS \in$ {CFS, Chi, Info}, for various learning methods $m \in$ {C4.5, SVM, NBay, MLP} and protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE}. We take the performance $S(M, 2)$ of best exhaustive search on each protein site $s$ and compare it to the best performance on the same protein site from feature selection methods $S(FS + m, 2)$ where $FS \in$ {CFS, Chi, Info} and $m \in \{C4.5, SVM, NBay, MLP\}$.

Table 4.40 shows the number of sites $s$ that achieve a $S(M, 2)$ measure, based on the best combination of multiple sets of micro-environment properties through an exhaustive search, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(FS + m, 2)$ for combination of feature selection method $FS \in$ {CFS, Chi, Info} and machine learning method $m \in \{C4.5, SVM, NBay, MLP\}$. Here we use implementation of WEKA package [65] with default settings. It is clear from the table that for all types of protein sites, the best combinations of multiple data sets give higher performance by $S(M, 2)$ than the best performance from conventional feature

selection methods.

Table 4.40: Number and percentage for EXH>FS, EXH=FS, and EXH<FS on 3 types of protein sites through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 3 | 3 | 3 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| % Sites | 100 | 100 | 100 | 67 | 0 | 0 | 0 | 33 | 0 | 0 | 0 | 0 |

### 4.5.3 Use of Selected Features on 26 Functions of Yeast Genes

In this section, we discuss classification studies, in which many more functions of yeast are involved than the earlier studies with 5 functions (as in Section 4.1.1). We compare the performance of selected features to that of using the best individual data sets and all available data sets. Also, we show how classification performance is improved when using a combination of whole data sets, instead of using features selected by conventional feature selection methods.

The focus of this section is to analyse the prediction accuracy of classification models using conventional feature selection methods—CFS (Correlation-based feature selection), Chi (Chi-squared feature selection), and Info (Information-gain feature selection)—and show their merits and demerits. In the previous subsection, we derived the $ALL$ data set by combining or merging 57 data sets of gene expression experiments (as described in Section 4.4.3) based on the 57 experimental conditions (different sets of features on same set of genes) (as detailed in Table 3.6). Now we build a classifier $C_f(ALL, FS + m)$ based on features selected by feature selection methods $FS$ using a machine learning method $m$ on a data set $E$. Here, the feature selection method $FS$ is applied on the training portion of the data to obtained a

reduced data set, the machine learning method $m$ is then applied to the reduced data set to obtain a classification model which is then applied to the testing portion of the data set.

Now, we apply conventional feature selection methods $FS \in \{\text{CFS, Chi, Info}\}$ on $ALL$ data set and build classification model $C_f(ALL, FS + m)$. In this section, we show only the performance of the $CFS$ method. Other results are shown in Appendix C. We show below in Table 4.41 the performance of $S(FS + m, 2)$ of the experiments just described. We can think of this table as the best performance by $S(M, 2)$ for predicting 26 functions of yeast using *FS*-selected data sets. The rows are for 26 functions annotated as per MIPS Catalogue (Version 2.0) of 19th March 2004, shown in column 1 as catalogue number and column 2 as function code. Column 3 shows the number of genes for each function. The fourth through seventh columns are the performance $S(FS + m, 2)$ through different learning algorithms $m \in \{C4.5, SVM, NBay, MLP\}$ from the WEKA package [65] with default settings, and $CFS$ is the feature selection method used.

Now, we compare $FS$ results with that of using the best individual data sets and with that of using all available data sets. We have the results of $C_f(E, m)$, where $m \in \{\text{C4.5, SVM, NBay, MLP}\}$, for 26 functions $f \in \{\text{Aam, Nsm, Nuc, Pho, Ccm, Lim, Vit, Tca, Res, Fer, Mer, Ecr, Dna, Cyc, Tcp, Rsn, Rpr, Rmo, Pro, Rib, Tra, Trc, Ami, Pft, Pfs, Ptt, Prm, Apc, Deg, Tcs, Tfc, Trt, Int, Rdv, Str, Dtx, Hom, Csr, Tvp, Gro, Dea, Wal, Cyt, Nuc, Mit, Fun}\}$ annotated as per MIPS Catalogue (Version 2.0) of 19th March 2004, and for various data sets $E \in \mathcal{E} = \{\text{Alp1, Alp2, Alp3, Alp4, Ace, Des1, Des2, Des3, Des4, Des5, Des6, Des7, Haa, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15,}$

Table 4.41: Performance by $S(M, 2)$ on 26 functions of yeast based on selected features through Correlation-based feature selection and algorithms.

| Function | Code | Genes | C4.5 | NBay | MLP | SVM |
|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | -76 | -390 | -20 | -10 |
| 11.04 | Rpr | 161 | -27 | -192 | -19 | -28 |
| 10.03 | Cyc | 149 | -60 | -227 | -40 | -8 |
| 20.09 | Trt | 145 | -28 | -452 | -53 | -1 |
| 12.01 | Rib | 138 | 203 | 184 | 223 | 223 |
| 1.01 | Aam | 103 | 42 | 2 | 55 | 56 |
| 1.06 | Lim | 99 | -1 | -49 | -15 | 10 |
| 10.01 | Dna | 99 | -30 | -196 | -16 | -1 |
| 1.05 | Ccm | 82 | -35 | -135 | -30 | -6 |
| 1.03 | Nuc | 81 | -10 | -79 | 4 | 3 |
| 14.13 | Deg | 77 | -12 | -167 | 13 | 11 |
| 32.01 | Str | 58 | -8 | -110 | 0 | -1 |
| 1.07 | Vit | 54 | 0 | -87 | -5 | 0 |
| 14.07 | Prm | 48 | 0 | -7 | 0 | 0 |
| 20.01 | Tcs | 46 | 4 | -57 | 1 | 0 |
| 12.04 | Tra | 42 | -22 | -199 | -15 | -8 |
| 11 | Tcp | 39 | 0 | -3 | 0 | 0 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | 5 | -75 | 8 | 7 |
| 20.03 | Tfc | 32 | 0 | -28 | 0 | 0 |
| 42.01 | Wal | 32 | 0 | -752 | -1 | 0 |
| 12.10 | Ami | 31 | 2 | -48 | 2 | 0 |
| 43.01 | Fun | 31 | 0 | -9 | 0 | 0 |
| 2.13 | Res | 29 | 5 | -78 | 11 | 8 |
| 14.01 | Pfs | 29 | -7 | -11 | 0 | 0 |
| 32.07 | Dtx | 27 | -6 | -34 | -8 | 0 |

Hea16, Hea17, Hea18, Hea19, Hea20, Dby1, Dby2, Dby3, Dby4, Dby5, Dby6, Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8, Fch1, Fch2, Met, Hyd, Iro, Aft, Fit, Pho, Snf, Spo} derived from experimental conditions (different sets of features on same set of genes). We get performance $S(M, 2)$ score based on the method $m$ on the best individual data sets; that is, for a function $f$, $S(M, 2) = \max_{E \in \mathcal{E}} C_f(E, m)$. We already got the performance for a function $f$, $S(FS{+}m, 2) = C_f(ALL, FS{+}m)$,where $FS \in$ {CFS, Chi, Info} (Note: WEKA does not support Fisher, T-test), for various learning methods $m \in$ {C4.5, SVM, NBay, MLP}, for 26 functions annotated as per MIPS Catalogue (Version 2.0) of 19th March 2004.

Table 4.42 shows the 26 functions that achieve a $S(M, 2)$ measure, based on the best individual data sets, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(FS + m, 2)$ for the combination of feature selection method $FS$ and machine learning method $m$. Here we use implementations in the WEKA package [65] with default settings. It is clear from the tables that for a many protein functions, a feature selection method gives a poorer performance by $S(M, 2)$ than that of using the best of individual data sets.

Table 4.42: Number and percentage for BI>FS, BI=FS, and BI<FS on 26 functions of yeast through algorithms. Number and percentage of .

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BI-FS | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| 26 Functions | 14 | 9 | 24 | 15 | 9 | 10 | 1 | 5 | 3 | 7 | 1 | 6 |
| % Functions | 54 | 35 | 92 | 58 | 35 | 38 | 4 | 19 | 12 | 27 | 4 | 23 |

Table 4.42 further shows that Best individual data sets yields greater performance by $S(M, 2)$ in 60% of the functions, equal in 24%, and lesser in 16% to FS data, over

4 methods.

We also have the results of $C_F(ALL, m)$, where $m \in$ {C4.5, SVM, NBay, MLP}, for 26 specific functions (as described in Section 4.4.3). We already got the performance for a function $F$, $S(FS+m, 2) = C_F(ALL, FS+m)$, where $FS \in$ {CFS, Chi, Info}, for various learning methods $m \in$ {C4.5, SVM, NBay, MLP}, for 26 functions annotated as per MIPS Catalogue (Version 2.0) of 19th March 2004.

Table 4.43 shows the number of functions that achieve a $S(M, 2)$ measure, based on using all available data sets, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(FS+m, 2)$ for the combination of feature selection method $FS$ and machine learning method $m$. Here we use implementations in the WEKA package [65] with default settings. It is clear from the table that for a majority functions, a feature selection method gives a better performance by $S(M, 2)$ than that of using all available data sets.

Table 4.43: Total number and percentage of 26 functions of yeast for ALL>FS, ALL=FS, and ALL<FS through algorithms.

| | Grt. $FS$ | | | | Eq. $FS$ | | | | Les. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| 26 Functions | 1 | 2 | 1 | 8 | 0 | 0 | 0 | 2 | 25 | 24 | 25 | 16 |
| % Functions | 4 | 8 | 4 | 31 | 0 | 0 | 0 | 8 | 96 | 92 | 96 | 62 |

Tables 4.43 further shows that ALL data sets yields greater performance by $S(M, 2)$ in 12% of the functions, equal in 2%, and lesser in 87% to FS data, over 4 methods.

We just provided the accuracy of inferring if a gene has one of the 26 functions based on a feature vector derived from gene expression profile of that gene under the

$FS$-selected data set. It is natural to speculate if one can perform better by using a feature vector derived by combining or merging gene expression profiles of that gene under two or more experimental conditions (different sets of features on same set of genes) but not all of them.

For 57 data sets, there are millions of combinations ($= 2^{57} - 59$) that involve more than 1 of these data sets but not all of them. This is a quite a large number of combinations. Discussions from the earlier sections also show that choosing combinations in a fixed but arbitrary order or in a random way do not give consistently good performance.

In Chapter 5, in Section 5.4.1 we will develop a Greedy-Hill climbing algorithm to pick a good combination of data sets. We show that for a majority of protein functions, the combinations chosen by the Greedy-Hill climbing algorithm give higher performance than using $ALL$ data set and also give a higher or equal performance than the best performance from feature selection methods.

## 4.6 Conclusion on Existing Methods

In the study of biological problems, many researchers tend to perform and add as many experimental assays as they can to their studies. However, some of these experiments may not be useful. Biologists need to have a good model to know which experiments are not very useful in a functional study. This helps them to better allocated limited resources. Machine learning methods are frequently applied on biological problems with an aim to maximize prediction accuracy of classification models.

In this chapter, we formulated a hypothesis that using pre-selected data sets in

classification studies is useful. We studied individual data set, additional data sets by selection and random methods, conventional feature selection methods, exhaustive search, combined data sets. We demonstrated the above by taking 3 specific examples:

1. 5 functions of yeast are studied through best individual data sets, exhaustive search, random sampling, and "add one data set at a time in a fixed order", $ALL$ data, $FS$-selected data and were compared with exhaustive search results.

2. 3 types of protein sites are studied with micro-environment properties surrounding protein sites by best individual micro-environment property, exhaustive search, random sampling, and "add one data set at a time in a fixed order", $ALL$ categories of micro-environment properties, $FS$-selected data and were compared with exhaustive search.

3. 26 cellular functions of yeast from MIPs functional annotation over 57 data sets by best individual data set, $ALL$ data set, $FS$-selected data, and some combinations of data sets chosen by a Greedy-Hill climbing algorithm (to be presented later in Chapter 5). We cannot use Exhaustive search in this problem due to millions of combinations out of 57 data sets that are used in our study.

The point to note is that previous researchers (Brown *et al.*, Mateos *et al.*, Bagley *et al.* and Wei *et al.*) use all available data sets together as one single combined data set in their classification studies. They did not show whether using all available data sets would consistently lead to better performance than using the best individual data sets. They also did not investigate the issue of the optimal choice of combinations of data sets. In this chapter, we demonstrated that the use of combined

data sets often lead to better performance than using the best individual data sets, using all data and selected by feature selection methods.

We demonstrated that the use of smaller combined data sets often lead to better performance than using all available data sets. This leads naturally to the problem: How to pick data sets that can significantly maximize classification performance? One traditional solution to this problem is to compute a "relevance" statistic—such as $\chi^2$, Student's t, etc.—on each individual features and picking the most relevant ones. We call this the "feature selection" approach.

Let us end this chapter with a technical note. Recall that we used the top "n" features to build a better classification model through the "Fisher" and "T-test" feature selection methods and the GIST package. As mentioned earlier, we "optimized" the choice of "n" by systematically trying out various values in a range and picked the best one ($n = 8$). This introduced a bias that gave an extra advantage to the performance of these conventional feature selection methods. However, as shown in this chapter, in spite of this bias in their favor, they did not achieve better accuracy than the use of a combination of whole data sets.

# Chapter 5

# Progressive Data Mining Through HILL and GREEDY-HILL

In Chapter 4, we demonstrated that the use of additional or combination of whole data sets through an exhaustive search—even when the data sets are derived from experiments (different sets of features on same set of genes or sites)—can increase prediction accuracy of classification models on a variety of bioinformatics problems. We cautioned that using all available data sets does not give the best improved prediction accuracy, and often gives a worse accuracy than using the best of individual data sets or the combination of whole data sets through an exhaustive search. We also showed how prediction accuracy can be improved by using selected features by conventional feature selection methods, compared to using the best of individual data sets or all available data sets. However, we also showed that using features selected by conventional feature selection methods does not achieve the best prediction accuracy often enough, compared to using a combination of whole data sets. In this chapter we show "Progressive Data Mining" (PDM) or Jumping Classifiers—a step-by-step performance escalating paradigm which enables selection of useful data sets among all available candidate data sets. PDM is different from leave-one-out or add-one-by-one

strategies in improving performance by choosing more data sets. We introduce two "whole dataset feature selection algorithms"—the "Hill climbing method" (*Hill*) and the "Greedy-Hill climbing method" (*Greedy-Hill*). Hill can handle a small number of data sets. Greedy-Hill can handle a larger number of data sets.

Specifically:

- We introduce the "Hill climbing method", Hill, for choosing a combination of whole data sets from a small number of data sets (up to 10) considered in a classification study. Hill chooses a new whole data set in each run to add to the already chosen ones to build a classification model with improved performance.

- We show here that the combination of whole data sets from Hill can yield better results, on predicting the 5 functions of yeast genes, than using the best of individual data sets, using all available data sets, and using selected features by conventional feature selection methods. Results show that for 60% (60%, 40%, and 60%, respectively) of the protein functional classes, we are able to use Hill to obtain a higher prediction accuracy than using the best of individual data sets through C4.5 (SVM, NBay, and MLP, respectively). Results also show that for 80% (80%, 60%, 60%, respectively) of the protein functional classes, we are able to use Hill to obtain a higher prediction accuracy than using all available data sets through C4.5 (SVM, NBay, and MLP, respectively). We also show for 40% (60%, 80%, and 40%) of the protein functional classes, we are able to use Hill to obtain a higher prediction accuracy than using selected features by conventional feature selection methods through C4.5 (SVM, NBay, and MLP, respectively). Furthermore, in almost all cases, the combination of whole data sets chosen by Hill is practically optimal the sense that they are as

good as the top 7% of the possible combinations of whole data sets, as verified by an exhaustive search.

- We show here that the combination of whole sets of micro-environment properties chosen by Hill yields better results, on the problem of protein sites prediction, than using the best of individual sets of micro-environment properties, using all available sets of micro-environment properties, and using selected sets of micro-environment properties by conventional feature selection methods. Results show that for 100% (100%, 67%, and 100%, respectively) of the protein sites, we are able to use Hill to obtain a higher prediction accuracy than using the best individual sets of micro-environment properties through C4.5 (SVM, NBay, and MLP, respectively). Results also show that for 67% (100%, 67%, 67%, respectively) of the protein sites, we are able to use Hill to obtain a higher prediction accuracy than using all available sets of micro-environment properties through C4.5 (SVM, NBay, and MLP, respectively). We also show for 100% (67%, 100%, and 33%) of the protein sites, we are able to use Hill to obtain a higher prediction accuracy than using micro-environment properties selected by conventional feature selection methods through C4.5 (SVM, NBay, and MLP, respectively). Furthermore, the combinations chosen by Hill are practically optimal in the sense that they are as good as the top 2% of the possible combinations of whole data sets, as verified by an exhaustive search.

- We then introduce the "Greedy-Hill climbing method", Greedy-Hill, for choosing a combination of whole data sets from a large number of available data sets (more than 10 sets) considered in a classification study. This method modifies Hill by allowing multiple data sets to be chosen in each cycle, as opposed

to the one-per-cycle strategy of Hill.

- We investigate the optimality of Greedy-Hill and its practicality on 5 functions of yeast and machine learning methods considered. We compare the combination of data sets chosen by Greedy-Hill to the performance of all possible combinations of data subsets in an exhaustive search, and also to the combination chosen by Hill. We show that a mere 3% of the possible combinations lead to better predictions than the combination chosen by Greedy-Hill, and 7.5% of the combinations chosen, over 4 learning methods, by Hill lead to better predictions than the combination chosen by Greedy-Hill. Thus Greedy-Hill is empirically within 3% of optimality. However, a typical round of model building using Greedy-Hill on a data set would take about 1367 seconds, over 4 learning method which is 1.263 times faster than the 1726 seconds, over 4 learning method, typically taken by Hill.

- We also applied Greedy-Hill on 3 types of protein sites and machine learning methods considered. We compare the combination of data subsets chosen by Greedy-Hill to the performance of all possible combinations of data subsets in an exhaustive search, and also to the combination chosen by Hill. We show that a mere 5% of the possible combinations lead to better predictions than the combination chosen by Greedy-Hill, and 8.3% of the combinations chosen, over 4 learning methods, by Hill lead to better predictions than the combination chosen by Greedy-Hill. Thus Greedy-Hill is empirically within 5% of optimality which is a little poorer than Hill. However, a typical round of model building using Greedy-Hill on a data set would take about 94 seconds, compared to 90 seconds by Hill, over 4 learning method.

- We show here that the combination of whole data sets chosen by Greedy-Hill achieves better results, on predicting 26 functions of yeast genes, than using the best of individual data sets, using all available data sets, and using selected features from conventional feature selection methods. Results show that for 30% (33%, 26%, and 43%, respectively) of the protein functional classes, we are able to use the combination of whole data sets chosen by Greedy-Hill to obtain a higher prediction accuracy than using the best of individual data sets through C4.5 (SVM, NBay, and MLP, respectively). Results also show that for 63% (83%, 91%, 76%, respectively) of the protein functional classes, we are able to use the combination of whole data sets chosen by Greedy-Hill to obtain a higher prediction accuracy than using all available data sets through C4.5 (SVM, NBay, and MLP, respectively). We also show that for 37% (43%, 72%, and 61%) of the protein functional classes, we are able to use the combination of whole data sets chosen by Greedy-Hill to obtain a higher prediction accuracy than using the best selected features by conventional feature selection methods through C4.5 (SVM, NBay, and MLP, respectively). We tabulate summary of performances for 26 functions of yeast and 20 functions of yeast through various methods.

- In Section 5.8 we discuss issues to further validate Progressive Data Mining performances through Multiple evaluation metrics. Also we compare PDM outcomes with that of Committee of Features, Committee Method, and 18 function through statistical sampling.

| D1 | D2 | D3 | D4 | D5 | D6 D56 | D57 |
|---|---|---|---|---|---|---|
| F1....F18 | F19..F24 | F... | F.... | F..... | F..... F..... | .F383..F390 |

Figure 5.1: Each data set:sets of experimental assays with biological time scale points.

## 5.1 Whole Dataset Feature Selection

Conventional feature selection methods generally select **individual features** to create proper boundaries between two classes based on the given data set. Conventional feature selection methods remove noisy features from the feature space, and hopefully achieve a higher classification accuracy.

In subsection 5.1.1, we introduce the idea of "whole data set" feature selection. Then, in subsection 5.1.2, a new algorithm for whole dataset feature selection—the "Hill climbing method" (Hill)—is developed to select the combination of whole data sets from all available data sets.

### 5.1.1 Whole Data Set

Let $\{D_1, D_2, D_3, ..., D_{57}\}$ be the data sets where $D_1$ consists features $\{F_1, ..., F_{18}\}$ and $D_2$ consists features $\{F_{19}, ..., F_{24}\}$, ..., $D_{57}$ consists features $\{F_{384}, ..., F_{390}\}$. Figure 5.1 shows that each data set: sets of experimental assays with biological time scale points. For example, $D_1$ is a data set with 18 features—$F_1$, $F_2$, ..., $F_{17}$, $F_{18}$—which are the time scale points from a wet experiment.

A conventional feature selection method treats each individual time scale point as one single feature. On the other hand, we treat a "whole individual data set" as a single "feature" that contains many sets of sub features or time scale points from a set of wet experiments. Our aim is to find a combination of whole data sets $\mathcal{C} \subseteq \{D_1, , ... D_n\}$, from all available data sets $\{D_1, ..., D_n\}$, for a better classification performance. We treat one whole data set—$D_1$ or $D_2$ or $D_3$ or ... or $D_{57}$—as one single "feature" selected by our whole data set feature selection methods.

### 5.1.2 The Hill Climbing Algorithm

The main idea of whole data set feature selection is to treat a **whole data set** as one feature, and to iteratively choose the best combination of whole data sets from all available data sets using a machine learning algorithm, with a performance check at each iteration. To evaluate the whole data set feature selection approach, we devise a simple Hill climbing method (Hill) for choosing data sets to learn from during the training phase. For Hill, at each iteration, we combine each available whole data set with the combination of whole data sets chosen at the previous iteration. At the end of an iteration, the data set that gives rise to the best performance is added to the chosen combination. The process is then repeated until no additional data set can be added to further increase performance. We apply Hill on 5 functions of yeast (Section 5.2.1) and 3 types of protein sites (Section 5.3.1). Let us describe Hill more formally. Let $D_{start}$ be a starting microarray data set that is to be analyzed with a classification learning algorithm $M \in \{$C4.5, SVM, NBay, MLP$\}$. Typically, $D_{start}$ would be a new microarray data set generated by one's own laboratory. We want to maximize the performance measure $S(M, 2)$, using a learning algorithm $M$ on this data set by combining it with additional data sets from other microarray studies

collected from different sources based on same gene or site. Let $\Phi_{additional} = \{D_1, \ldots, D_n\}$ be $n$ additional microarray data sets conducted on the same set of genes as $D_{start}$. These additional data sets are from different laboratories through various experimental studies. Our objective is to search for a subset of $\Phi_{additional}$ which can be combined with $D_{start}$ to give the best data analysis results by $M$:

**Step 1 (Optional):** Normalize the expression vectors in $D_{start}, D_1, \ldots, D_n$ so that they each have mean 0 and standard deviation 1. Note that other appropriate normalization methods can also be used.

**Step 2:** Let $\mathcal{S}_{M,\Psi}$ be the $S(M,2)$ score of applying $M$ on the data sets in $\Psi \subseteq \Phi_{all}$, where $\Phi_{all} = \Phi_{additional} \cup \{D_{start}\}$. Set $\mathcal{D}_{best}^{(0)} := D_{start}$.

**Step 3:** In the $i$-th iteration, find

$$\mathcal{D}_{best}^{(i)} = \text{argmax}_{D_j \in \Phi_{all} - \{\mathcal{D}_{best}^{(k)} \mid 0 \le k \le i-1\}} \mathcal{S}_{M,\{\mathcal{D}_{best}^{(k)} \mid 0 \le k \le i-1\} \cup \{D_j\}}$$

$$S_{best}^{(i)} = S_{M,\{D_{best}^{(k)} \mid 0 \le k \le i\}}$$

**Step 4:** Halt the iteration process in Step 3 if $i > n$ or $\mathcal{S}_{best}^{(i)} \le \mathcal{S}_{best}^{(i-1)}$.

Upon termination, $\mathcal{D}_{best}^{(1)}$, ..., $\mathcal{D}_{best}^{(i-1)}$ will be a selection of additional microarray data sets that can be combined with $D_{start}$ to produce a better classification performance than that from just using $D_{start}$ alone. In Section 5.2.1 and Section 5.3.1 we report results from applying Hill on 5 functions of yeast and 3 types of protein sites.

## 5.2 Inferring 5 Specific Functions of Yeast Genes

In Subsection 5.2.1, we present the performance by $S(M,2)$ of Hill on inferring the 5 specific functions of yeast genes from 6 sets of gene expression experiments.

In Subsection 5.2.2, we compare the performance by $S(M, 2)$ of using the combination of whole data sets chosen by Hill to that of using the best of individual data sets, using all available data sets, and using selected features by conventional feature selection methods.

In Subsection 5.2.3, we show that classification performance by $S(M, 2)$ of the combination of whole data sets chosen by Hill is practically optimal by comparing it with an exhaustive search through all possible combinations of whole data sets.

### 5.2.1 The Study of 5 Specific Functions of Yeast Genes Using Hill Chosen Data Sets

The focus of this chapter is to analyse the prediction accuracy of classification models using the combination of whole data sets chosen by Hill—and show their merits and demerits. To build a classifier $C_f(C, Hill + m)$ based on the combination of whole data sets chosen by Hill for a machine learning method $m$ on a collection $C$ of data sets, we do the followings. Hill is applied on the training portion of $C$ to obtained a reduced data set. The machine learning method $m$ is then applied to the reduced data set to obtain a classification model. The model is then applied to the testing portion of $C$.

Now, we apply Hill on the collection of data sets $\mathcal{A} = \{$Alp, cdc, Elu, ccc, Spo, Dia$\}$, and build a classification model $C_f(\mathcal{A}, Hill + m)$, for $m \in \{$C4.5, SVM, NBay, MLP$\}$ and functions $f \in \{$HIST, PROT, RESP, RIBO, TCA$\}$. We show below in Table 5.1 the results of the experiments just described, using SVM and MLP (we only show SVM and MLP due to space constraint) as the learning methods with Hill chosen data. We can think of this table as the performance by $S(M, 2)$ through SVM and MLP for predicting 5 specific cellular functions of yeast genes using Hill

chosen data. The rows are the 5 functions—HIST (histones), PROT (proteosomes), RESP (respiration), RIBO (ribosomes), and TCA (TCA cycle). Column 2 shows the performance $S(Hill + SVM, 2)$ using the combination of whole data sets chosen by Hill for SVM. Column 3 shows the combination of whole data sets chosen by Hill for SVM in the format $\{D_1 + D_2 + .... + D_n\}$. Column 4 shows the performance $S(Hill + MLP, 2)$ using the combination of whole data sets chosen by Hill for MLP. Column 5 shows the combination of whole data sets chosen by Hill for MLP in the format $\{D_1 + D_2 + .... + D_n\}$. The SVM here is the support vector machine implementation from the GIST package (Release 2.0.5, April 30, 2003) that uses RBF of degree 3 as in Brown *et al.* The MLP here is the multilayer perceptron implementation from the WEKA package [65] at its default settings. 1,851 genes in 60 non-singleton functional classes are annotated based on the MIPS Catalogue (Version 1.3) of 25th June 2003.

Table 5.1: Performance by $S(M, 2)$ on 5 functions of yeast based on selected combination of data sets by Hill through SVM and MLP.

| Function | SVM | | MLP | |
|---|---|---|---|---|
| | $S(Hill + SVM, 2)$ | data sets | $S(Hill + MLP, 2)$ | data sets |
| HIST | 18 | $Cdc + Spo$ | 18 | $Elu$ |
| PROT | 39 | $Cdc + Ccc + Spo$ | 40 | $Spo + Dia$ |
| RESP | -10 | $Cdc + Elu + Spo$ | 0 | $Alp$ |
| RIBO | 227 | $Cdc + Spo + Dia$ | 200 | $Elu + Spo$ |
| TCA | 8 | $Alp + Elu + Spo$ | 0 | $Ccc$ |

### 5.2.2 Comparison of Hill Chosen Data to Best of Individual Data Sets, All Available Data Sets, and Selected Features

In the previous subsection we showed the accuracy of inferring if a gene has one of the 5 specific functions—HIST, PROT, RESP, RIBO, or TCA—based on a feature vector derived from the gene expression profile of that gene by Hill.

In this subsection, we compare Hill results with that of using the best of individual data sets, all available data sets, and selected features by conventional feature selection methods.

In Brown *et al.*, there are 8 data sets corresponding to 8 experimental conditions (different sets of features on same set of genes)—$\mathcal{A}$ = {Alp, Cdc, Elu, Spo, Dia, HEAT, DTT, COLD}. We got the performance for a function $f$, $S(M, 2) = \max_{C \in \mathcal{A}} C_f(C, m)$, using the best of individual data sets and various learning methods $m \in$ {C4.5, SVM, NBay, MLP}, for a function $f \in$ {HIST, PROT, RESP, RIBO, TCA}. We also got the performance $S(Hill + m, 2)$ for a function $f$ from $C_f(\mathcal{A}, Hill + m)$. Table 5.2 shows the number of functions $f$ that achieve a $S(M, 2)$ measure, based on the best of individual data sets, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(Hill + m, 2)$ for the combination of whole data sets chosen by Hill for machine learning method $m$. Here, we use implementations in the WEKA package [65] with default settings. It is clear from the table that, for most protein functions, Hill gives a higher performance by $S(M, 2)$ than using the best of individual data sets.

We calculate the performance by $S(M, 2)$ for a function $f$ from $C_f(ALL, m)$, where $ALL$ is a data set derived by combining or merging 8 data sets—Alp, Cdc, Elu, Spo,

Table 5.2: Number and percentage for BI>Hill, BI=Hill, and BI<Hill on 5 functions of yeast through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 0 | 0 | 0 | 0 | 2 | 2 | 3 | 2 | 3 | 3 | 2 | 3 |
| % Functions | 0 | 0 | 0 | 0 | 40 | 40 | 60 | 40 | 60 | 60 | 40 | 60 |

Dia, HEAT, DTT, and COLD based on same gene. We also got the performance $S(Hill+m, 2)$ for a function $f$ from $C_f(\mathcal{A}, Hill+m)$. Our annotations are based on the MIPS Catalogue (Version 1.3) of 25th June 2003. Table 5.3 shows the number of functions $f$ that achieve a $S(M, 2)$ measure, based on using all the available data sets, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(Hill+m, 2)$ for the combination of whole data sets chosen by Hill for machine learning method $m$. Here, we use implementations in the WEKA package [65] with default settings. It is clear from the table that, for almost all protein functions, Hill gives a higher performance by $S(M, 2)$ than using all available data sets.

Table 5.3: Number and percentage for ALL>Hill, ALL=Hill, and ALL<Hill on 5 functions of yeast through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 2 | 4 | 4 | 3 | 3 |
| % Functions | 20 | 20 | 20 | 0 | 0 | 0 | 20 | 40 | 80 | 80 | 60 | 60 |

We also got the performance $S(FS+m, 2)$ for a function $f$ from $C_f(ALL, FS+m)$, for conventional feature selection methods $FS \in$ {CFS, Chi, Info}, learning methods $m \in$ {C4.5, SVM, NBay, MLP}, and functions $f \in$ {HIST, PROT, RESP, RIBO,

TCA}. We also got the performance $S(Hill+m, 2)$ for a function $f$ from $C_f(\mathcal{A}, Hill+m)$. Our annotations are based on the MIPS Catalogue (Version 1.3) of 25th June 2003. Table 5.4 shows the number of functions $f$ that achieve a $S(M, 2)$ measure, based on selected features by conventional feature selection methods, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(Hill + m, 2)$ for the combination of whole data sets chosen by Hill for machine learning method $m$. Here, we use implementations in the WEKA package [65] with default settings. It is clear from the table that, for most protein functions, Hill gives a higher performance by $S(M, 2)$ than using selected features by conventional feature selection methods.

Table 5.4: Number and percentage for FS>Hill, FS=Hill, and FS<Hill on 5 functions of yeast through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 2 | 1 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 3 | 4 | 2 |
| % Functions | 40 | 20 | 20 | 20 | 20 | 20 | 0 | 40 | 40 | 60 | 80 | 40 |

Table 5.5 summarizes the performance by $S(M, 2)$ of using the best of individual data sets, using all available data sets, using selected features by conventional feature selection methods, by using the combination of whole data sets chosen by Hill. The results here are achieved by using the 8 data sets corresponding to 8 experimental conditions (different sets of features on same set of genes)—Alp, Cdc, Elu, Spo, Dia, HEAT, DTT, COLD, from Brown *et al.*—with annotations based on MIPS Catalogue (Version 1.3) of 25th June 2003.

The rows are the 5 functions—HIST (histones), PROT (proteosomes), RESP

(respiration), RIBO (ribosomes), and TCA (TCA cycle). Column 2 shows different machine learning algorithms $m \in \{C4.5, SVM, NBay, MLP\}$ used in the study. Column 3 shows the performance $S(M, 2)$ for a function $f$ using the best of individual data sets. Column 4 shows the performance $S(ALL + m, 2)$ for a function $f$ using all available data sets. Column 5 shows the performance $S(FS + m, 2)$ for a function $f$ using selected features by conventional feature selection methods. Column 6 shows the performance of $S(Hill + m, 2)$ on a function $f$ using the combination of whole data sets chosen by Hill. Column 7 shows the performance $S(Exh + m, 2)$ on a function $f$ using the best combination of whole data sets through an exhaustive search.

Table 5.5 shows some exceptional results. For example, FS got higher outcome than Hill and EXH. Also Hill got higher outcome than EXH. Conventional feature selection selects individual features for a model. On the other hand, Hill or EXH chooses combination of whole data sets and not combination of individual features. The explanation on how Hill got better outcome than EXH is detailed at the end of the Section 5.2.

### 5.2.3 Using Hill Chosen Data Improves Prediction Accuracy on 5 Functions of Yeast Genes

In Subsection 5.2.1 we showed the accuracy of inferring if a gene has one of the 5 specific functions—HIST, PROT, RESP, RIBO, or TCA—based on a feature vector derived by Hill from the gene expression profile of that gene. Since we have shown in previous chapter that using the combination of whole data sets through an exhaustive search can improve prediction performance, it is natural to compare the best combination of whole data sets through an exhaustive search to the combination of whole data sets chosen by Hill. In this subsection, we generate all combinations of 2

Table 5.5: Performance by $S(M, 2)$ of 5 cellular functions of yeast using the best of individual data sets, using all available data sets, best performance from conventional feature selection methods, using the combination of whole data sets chosen by Hill, and using the best combination of whole data sets through an exhaustive search.

| 5 Function | ALG | S(Best-Ind) | S(ALL) | S(Best-FS) | S(Hill) | S(Best-Exh) |
|---|---|---|---|---|---|---|
| HIST | C4.5 | 12 | 13 | 13 | 14 | 15 |
| PROT | C4.5 | 15 | 29 | 41 | 33 | 33 |
| RESP | C4.5 | 0 | -8 | 0 | 0 | 0 |
| RIBO | C4.5 | 174 | 192 | 195 | 216 | 213 |
| TCA | C4.5 | 0 | 6 | 11 | 0 | 6 |
| HIST | SVM | 16 | 15 | 14 | 16 | 16 |
| PROT | SVM | 0 | 31 | 35 | 24 | 38 |
| RESP | SVM | 0 | -32 | 0 | 0 | 0 |
| RIBO | SVM | 160 | 223 | 221 | 232 | 233 |
| TCA | SVM | 0 | 3 | 7 | 8 | 7 |
| HIST | NBay | 13 | 16 | 14 | 16 | 16 |
| PROT | NBay | 1 | 19 | 39 | 1 | 21 |
| RESP | NBay | 0 | -132 | -27 | 0 | 0 |
| RIBO | NBay | 174 | 211 | 212 | 227 | 228 |
| TCA | NBay | 0 | -37 | -34 | 0 | 0 |
| HIST | MLP | 16 | 16 | 16 | 16 | 16 |
| PROT | MLP | 31 | 33 | 38 | 39 | 40 |
| RESP | MLP | 0 | -24 | 0 | 0 | 2 |
| RIBO | MLP | 189 | 213 | 219 | 230 | 235 |
| TCA | MLP | 3 | 4 | 12 | 4 | 12 |

or more data sets, and compare the performance from the best combination of whole data sets through an exhaustive search to the performance from the combination of whole data sets chosen by Hill.

Recall that for Brown *et al.* there are 8 data sets corresponding to 8 experimental conditions (different sets of features on same set of genes). This yields 255 ($= 2^8 - 1$) non-empty sets that involve the 8 data sets. We have a sufficiently small number of combination of whole data sets here. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in \{$C4.5, SVM, NBay, and MLP$\}$ for predicting the 5 specific functions $f \in \{$HIST, PROT, RESP, RIBO, TCA$\}$ using these 255 sets of data sets, and compare the best performance on the same protein function using Hill chosen data. Results of this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 5.6, 5.8, 5.7, and 5.9. In each of these 4 tables, the second, third, and fourth columns show the number of sets that give poorer, equal, and better performance than the performance $S(Hill + m, 2)$. The fifth, sixth, and seventh columns show the respective percentages with respect to the 255 total possible sets.

Table 5.6: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 5 functions of yeast through SVM.

| Function | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| HIST | 245 | 10 | 0 | 96.08 | 3.92 | 0.00 |
| PROT | 228 | 6 | 21 | 89.41 | 2.35 | 8.24 |
| RESP | 203 | 52 | 0 | 79.61 | 20.39 | 0.00 |
| RIBO | 251 | 3 | 1 | 98.43 | 1.18 | 0.39 |
| TCA | 255 | 0 | 0 | 100.00 | 0.00 | 0.00 |

From these 4 tables we can see that out of the 5100 combination of data sets

Table 5.7: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 5 functions of yeast through NBay.

| Function | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| HIST | 253 | 2 | 0 | 99.22 | 0.78 | 0.00 |
| PROT | 227 | 2 | 26 | 89.02 | 0.78 | 10.20 |
| RESP | 252 | 3 | 0 | 98.82 | 1.18 | 0.00 |
| RIBO | 254 | 0 | 1 | 99.61 | 0.00 | 0.39 |
| TCA | 254 | 1 | 0 | 99.61 | 0.39 | 0.00 |

Table 5.8: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 5 functions of yeast through C4.5.

| Function | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| HIST | 248 | 6 | 1 | 97.25 | 2.35 | 0.39 |
| PROT | 253 | 2 | 0 | 99.22 | 0.78 | 0.00 |
| RESP | 137 | 118 | 0 | 53.73 | 46.27 | 0.00 |
| RIBO | 255 | 0 | 0 | 100.00 | 0.00 | 0.00 |
| TCA | 230 | 14 | 11 | 90.20 | 5.49 | 4.31 |

Table 5.9: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 5 functions of yeast through MLP.

| Function | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| HIST | 236 | 19 | 0 | 92.55 | 7.45 | 0.00 |
| PROT | 253 | 1 | 1 | 99.22 | 0.39 | 0.39 |
| RESP | 239 | 15 | 1 | 93.73 | 5.88 | 0.39 |
| RIBO | 238 | 4 | 13 | 93.33 | 1.57 | 5.10 |
| TCA | 239 | 6 | 10 | 93.73 | 2.35 | 3.92 |

through an exhaustive search over SVM, C4.5, NBay, and MLP on the 5 specific cellular functions, 264 (=5%) of the possible combinations of whole data sets yield an accuracy equal to Hill chosen data, 4750 (=93%) of the possible combinations of whole data sets yield lesser accuracy than Hill chosen data, and 86 (=2%) of total combinations of data sets yield better accuracy than Hill chosen data. This means that the combination chosen by Hill is among the 7% of combinations that give the best performance, at least for the purpose of predicting the 5 specific functions of yeast genes.

Table 5.10 shows the number of functions $f$ that achieve a $S(M, 2)$ measure, based on the best combinations of whole data sets through an exhaustive search, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(Hill + m, 2)$. Here, we use implementations from the WEKA package [65] with default settings. It is clear from the table that, for a majority of protein functions, Hill achieves as good a performance as the best performance through an exhaustive search.

Table 5.10: Total number and percentage of 5 functions of yeast for EXH>Hill, EXH=Hill, and EXH<Hill through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 2 | 2 | 2 | 4 | 2 | 2 | 3 | 1 | 1 | 1 | 0 | 0 |
| % Functions | 40 | 40 | 40 | 80 | 40 | 40 | 60 | 20 | 20 | 20 | 0 | 0 |

Also, we calculate the average $S(M, 2)$ performances—using the best of individual data sets (BI), using all available data sets (ALL), using selected features from conventional feature selection methods (FS), and using Hill—over 4 machine learning methods. Results from these approaches are BI:40.2, ALL:39.55, FS:51.3, and

Hill:53.8. For comparison, the average $S(M, 2)$ performance through the exhaust search is Exh:56.55. Thus, Hill has the best average performance by $S(M, 2)$, amongst the non-exhaustive-search methods, for the purpose of predicting 5 functions of yeast.

Finally, a remark on Table 5.5. Notice that for RIBO, S(Hill) = 216 and S(Best-Exh) = 213. This is quite unexpected and deserves a short discussion. Recall that S(Best-Exh) is supposed to be the score achieved by the best combination through an exhaustive search. Since the combination chosen by Hill must be one of those combinations explored also by the exhaustive search, S(Hill) should never be greater than S(Best-Exh). So what is happening? It turns out that many machine learning algorithms, including C4.5, are slightly sensitive to the ordering of features in the data sets. For example, for the function RIBO using C4.5, S(Best-Exh) = S(Hill) = 216 using the ordering {DTT, Spo, Dia, Cold}, but S(Best-Exh) = 206 using the ordering {Spo, DTT, Cold, Dia}. As another example, for the function TCA using SVM, S(Best-Exh) = S(Hill) = 8 using the ordering {Elu, Dia, Alp}, but S(Best-Exh) = $-8$ using the ordering {Alp, Elu, Dia}. We did not attempt to search through all orderings exhaustively, as the number of orderings is explosively large even for a small number of data sets.

## 5.3 Inferring Protein Sites

In Subsection 5.3.1, we present the performance by $S(M, 2)$ of Hill on inferring the 3 types of protein sites.

In Subsection 5.3.2, we compare the performance by $S(M, 2)$ of using the combination of whole micro-environment properties to that of using the best of individual

data sets, all available data sets, and micro-environment properties selected by conventional feature selection methods.

In Subsection 5.3.3, we show that the classification performance by $S(M, 2)$ of using the combination of whole data sets of micro-environment properties chosen by Hill is as good as the best combinations that can be found using an exhaustive search.

### 5.3.1 The Study of 3 Specific Protein Sites Using Hill Chosen Micro-Environment Properties

The focus of this chapter is to analyse prediction accuracy of classification models using the combination of whole sets of micro-environment properties chosen by Hill—and show their merits and demerits. We have 6 sets of micro-environment properties—$\mathcal{A}$ = {Atom, Chemical, Residue, Sec-str, Others, Co-ord}. To build a classifier $C_s(\mathcal{A}, Hill{+}m)$ based on the combination of whole sets of micro-environment properties chosen by Hill for a machine learning method $m$ on $\mathcal{A}$, we do the following. Hill is applied on the training portion of $\mathcal{A}$ to obtained a reduced data set, the machine learning method $m$ is then applied to the reduced data set to obtain a classification model, which is then applied to the testing portion of $\mathcal{A}$.

Now, we apply Hill on the collection of data sets $\mathcal{A}$ = {Atom, Chemical, Residue, Sec-str, Others, Co-ord}, and build a classification model $C_s(\mathcal{A}, Hill + m)$, for learning method $m \in$ {C4.5, SVM, NBay, MLP} and protein sites $s \in$ {CALCIUM, DISULFIDE, SERINE}. We show below in Table 5.11 the results of the experiments just described. We can think of this table as the performance by $S(M, 2)$ by 4 learning algorithms for predicting 3 types of protein sites using Hill chosen data. The rows are the 3 types of protein sites—CALCIUM (calcium binding sites), SERINE (serine protease active sites), and DISULFIDE (disulfide bridges). The second

through the fifth columns are the performance $S(Hill + C4.5, 2)$, $S(Hill + SVM, 2)$, $S(Hill + NBay, 2)$, and $S(Hill + MLP, 2)$, respectively. C4.5, SVM, NBay, and MLP are the implementations from the WEKA package [65] at its default settings.

Table 5.11: Performance by $S(M, 2)$ on 3 types of protein sites based on selected combination of micro-environment properties by Hill through algorithms.

| Prot-sites | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|
| CALCIUM | 185 | 169 | 157 | 183 |
| DISULFIDE | 68 | 64 | 53 | 65 |
| SERINE | 86 | 76 | 79 | 85 |

### 5.3.2 Comparison of Hill Chosen Data to Best of Individual Data Sets, All Available Data Sets, and Selected Features

In the previous subsection we gave the accuracy of inferring if an atom has one of the 3 types of protein sites—CALCIUM, SERINE, and DISULFIDE—based on Hill chosen data. In this subsection, we compare Hill results to that of using the best of individual data sets, all available data sets, and selected features by conventional feature selection methods.

We have 6 data sets corresponding to 6 micro-environment properties—$\mathcal{A}$ = {Atom, Chemical, Residue, Sec-str, Others, Co-ord}. We got the best performance for a site $s$ as $S(M, 2) = \max_{D \in \mathcal{A}} C_s(D, m)$ using the best of individual sets of micro-environment properties, for various learning methods $m \in$ {C4.5, SVM, NBay, MLP} and protein sites $s \in$ {CALCIUM, DISULFIDE, SERINE}. We also got performance $S(Hill + m, 2)$ for a protein site $s$ from $C_s(\mathcal{A}, Hill + m)$. Table 5.12 shows the number of protein sites $s$ that achieve a $S(M, 2)$ measure, based on the best of individual

sets of micro-environment properties, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(Hill + m, 2)$ for the combination of whole sets of micro-environment properties chosen by Hill for machine learning method $m$. Here, we use implementations in the WEKA package [65] with default settings. It is clear from the table that, for almost all protein sites, Hill gives a higher performance by $S(M, 2)$ than using the best of individual sets of micro-environment properties.

Table 5.12: Number and percentage for BI>Hill, BI=Hill, and BI<Hill on 3 types of protein sites through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 3 | 3 | 2 | 3 |
| % Sites | 0 | 0 | 0 | 0 | 0 | 0 | 33 | 0 | 100 | 100 | 67 | 100 |

We calculate the performance $S(M, 2)$ for a protein site $s$ from $C_s(ALL, m)$, where $ALL$ is a data set derived by combining or merging 6 data sets—Atom, Chemical, Residue, Sec-str, Others, and Co-ord based on same site. We also got the performance $S(Hill+m, 2)$ for a protein site $s$ from $C_s(\mathcal{A}, Hill+m)$. Table 5.13 shows the number of protein sites $s$ that achieve a $S(M, 2)$ measure, based on using all available micro-environment properties, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance $S(Hill + m, 2)$ for the combination of whole sets of micro-environment properties chosen by Hill for machine learning method $m$. Here, we use implementations from the WEKA package [65] with default settings. It is clear from the table that, for all protein sites, Hill gives a higher performance by $S(M, 2)$ than using all available micro-environment properties.

Table 5.13: Number and percentage for ALL>Hill, ALL=Hill, and ALL<Hill on 3 types of protein sites through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 2 | 3 | 2 | 2 |
| % Sites | 33 | 0 | 33 | 33 | 0 | 0 | 0 | 0 | 67 | 100 | 67 | 67 |

We got performance $S(FS + m, 2)$ for a site $s$ from $C_s(ALL, FS + m)$, for conventional feature selection methods $FS \in$ {CFS, Chi, Info}, for various learning methods $m \in$ {C4.5, SVM, NBay, MLP}, and for sites $s \in$ {CALCIUM, DISULFIDE, SERINE}. We also got performance $S(Hill + m, 2)$ from $C_s(\mathcal{A}, Hill + m)$. Table 5.14 shows the number of protein sites $s$ that achieve a $S(M, 2)$ measure, based on selected features by conventional feature selection methods, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(Hill + m, 2)$ for the combination of whole sets of micro-environment properties chosen by Hill for machine learning method $m$. Here, we use implementations in the WEKA package [65] with default settings. It is clear from the table that, for a majority of protein sites, Hill gives a higher performance by $S(M, 2)$ than using the selected micro-environment properties by conventional feature selection methods.

Table 5.14: Number and percentage for FS>Hill, FS=Hill, and FS<Hill on 3 types of protein sites through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 3 | 2 | 3 | 1 |
| % Sites | 0 | 33 | 0 | 33 | 0 | 0 | 0 | 33 | 100 | 67 | 100 | 33 |

Table 5.15 summarizes the performance by $S(M, 2)$ of using the best of individual

sets of micro-environment properties, using all available sets of micro-environment properties, using selected features by conventional feature selection methods, and using the best combination of whole sets of micro-environment properties chosen by Hill. The rows are the 3 types of protein sites—CALCIUM (calcium binding sites), DISULFIDE (disulfide bridges) and SERINE (serine protease active sites). Column 2 shows various machine learning algorithms $m \in \{C4.5, SVM, NBay, MLP\}$ used in the study. Column 3 shows the performance $S(M, 2)$ for a protein site $s$ using the best individual sets of micro-environment properties. Column 4 shows the performance $S(ALL+m, 2)$ for a protein site $s$ using all available sets of micro-environment properties. Column 5 shows the performance $S(FS + m, 2)$ for a protein site $s$ using selected features by conventional feature selection methods. Column 6 shows the performance $S(Hill + m, 2)$ on a protein site $s$ using the combination of whole sets of micro-environment properties chosen by Hill. Column 7 shows the best performance $S(Exh + m, 2)$ on a protein site $s$ using the combination of whole sets of micro-environment properties through an exhaustive search.

### 5.3.3 Using Hill Chosen Data Improves Prediction Accuracy on 3 Specific Types of Protein Sites

In the Subsection 5.3.1, we showed the accuracy of inferring if an atom has one of the 3 types of protein sites—CALCIUM, SERINE, and DISULFIDE—based on a feature vector derived by Hill from micro-environment profile of that atom. Since we have shown in previous chapter that using the combination of whole sets of micro-environment properties through an exhaustive search can improve the prediction performance by $S(M, 2)$, it is natural to compare the best combination of whole sets through an exhaustive search to the combination of whole sets of micro-environment

Table 5.15: Performance by $S(M, 2)$ of 3 types of protein sites using the best sets of micro-environment properties, using all available sets of micro-environment properties, best performance from conventional feature selection methods, using the best combination of whole sets of micro-environment properties selected by Hill, and using the best combination of whole sets of micro-environment properties through exhaustive search.

| Sites | ALG | S(Best-Ind) | S(ALL) | S(Best-FS) | S(Hill) | S(Best-Exh) |
|---|---|---|---|---|---|---|
| CALCIUM | C4.5 | 179 | 181 | 184 | 185 | 185 |
| SERINE | C4.5 | 80 | 66 | 83 | 86 | 86 |
| DISULFIDE | C4.5 | 58 | 83 | 67 | 68 | 73 |
| CALCIUM | SVM | 127 | 153 | 165 | 169 | 169 |
| SERINE | SVM | 15 | 66 | 65 | 76 | 76 |
| DISULFIDE | SVM | 6 | 60 | 65 | 64 | 66 |
| CALCIUM | NBay | 136 | 154 | 151 | 157 | 157 |
| SERINE | NBay | 72 | 45 | 78 | 79 | 79 |
| DISULFIDE | NBay | 53 | 65 | 44 | 53 | 53 |
| CALCIUM | MLP | 171 | 167 | 182 | 183 | 183 |
| SERINE | MLP | 80 | 68 | 85 | 85 | 85 |
| DISULFIDE | MLP | 64 | 71 | 67 | 65 | 68 |

properties chosen by Hill. In this subsection, we generate all combinations of 2 or more whole sets of micro-environment properties, and compare the performance by $S(M, 2)$ from the best combination of whole sets of micro-environment properties through an exhaustive search to the performance from the combination of whole sets of micro-environment properties chosen by Hill.

Recall that for annotated atoms there are 6 data sets. This yields 63 (= $2^6$ -1) non-empty sets. This is a sufficiently small number of sets. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 3 types of protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE} using these 63 sets of sets, and compare the best performance on the same protein site with Hill chosen data. Results of this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 5.16, 5.17, 5.18, and 5.19. In each of these 4 tables, the second, third, and fourth columns show the number of sets that give poorer, equal, and better performance than $S(Hill + m, 2)$. The fifth, sixth, and seventh columns show the respective percentages with respect to the 63 total possible sets.

Table 5.16: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 3 types of protein sites through SVM.

| Prot-site | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| DISULFIDE | 61 | 1 | 1 | 96.825 | 1.587 | 1.587 |

From these 4 tables, we can see that out of the 756 sets through an exhaustive search over SVM, C4.5, NBay, and MLP on the 3 specific types of protein sites, 16(=2%) of the possible combinations of whole sets of micro-environment properties

Table 5.17: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 3 types of protein sites through NBay.

| Prot-site | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| DISULFIDE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |

Table 5.18: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 3 types of protein sites through C4.5.

| Prot-site | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| DISULFIDE | 60 | 2 | 1 | 95.238 | 3.175 | 1.587 |

Table 5.19: Number and percentage for EXH<Hill, EXH=Hill, and EXH>Hill on 3 types of protein sites through MLP.

| Prot-site | Les. Hill | Eq. Hill | Grt. Hill | Les. Hill% | Eq. Hill% | Grt. Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| DISULFIDE | 58 | 4 | 1 | 92.063 | 6.349 | 1.587 |

yield an accuracy equal to Hill chosen data, 737 (=98%) of the possible combinations of whole sets of micro-environment properties yield lesser accuracy than Hill chosen data, and 3 (=0%) of total combinations of sets of micro-environment properties yield better accuracy than Hill chosen data. This means that the combination chosen by Hill is within the 2% of combinations that give the best performance. Hence, Hill is practically optimal, at least for the purpose of predicting the 3 specific types of protein sites.

Table 5.20 shows the number of protein sites $s$ that achieve a $S(M, 2)$ measure, based on the best combination of whole sets of micro-environment properties through an exhaustive search, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance of $S(Hill + m, 2)$ for the combination of whole sets of micro-environment properties chosen by Hill for machine learning method $m$. Here, we use implementations from the WEKA package [65] with default settings. It is clear from the table that, for a majority of protein sites, Hill achieves as good a performance as the best performance through an exhaustive search.

Table 5.20: Total number and percentage of 3 types of protein sites for EXH>Hill, EXH=Hill, and EXH>Hill through algorithms.

| | Grt. *Hill* | | | | Eq. *Hill* | | | | Les. *Hill* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Sites | 1 | 1 | 0 | 1 | 2 | 2 | 3 | 2 | 0 | 0 | 0 | 0 |
| % Sites | 33 | 33 | 0 | 33 | 67 | 67 | 100 | 67 | 0 | 0 | 0 | 0 |

Finally, we calculate the average $S(M, 2)$ performance by $S(M, 2)$—using the best set of micro-environment properties (BI), using all available sets of micro-environment

properties (ALL), using selected micro-environment properties by conventional feature selection methods (FS), and using the best combination of whole sets of micro-environment properties chosen by Hill, over 4 machine learning methods. Results from these approaches are BI:86.75, ALL:98.25, FS:103.00, and Hill:105.83. For comparison, the average performance by $S(M, 2)$ from exhaustive search is Exh: 106.67. So Hill achieves the best average performance amongst the non-exhaustive-search methods, for the purpose of predicting 3 specific types of protein sites.

## 5.4 Greedy-Hill Climbing Method

In Section 5.1, we devised a simple Hill climbing method, Hill, for choosing the best combination of whole data sets. In Section 5.2.1, application of Hill on 5 functions of yeast and in Section 5.2.2, application of Hill on 3 types of protein sites, are shown. We also got the best combination of whole data sets through an exhaustive search in Subsection 4.2.1 out of 255 sets on 5 functions of yeast and in Subsection 4.2.2 out of 63 sets on 3 types of protein sites. Both were feasible because of the small number of combinations of whole data sets involved. If Hill or an exhaustive search were applied on 26 functions of yeast with 57 data sets, it would become computationally expensive $(2^{57} - 1)$, due to the millions of combinations arising out of all available data sets. So far, Hill has achieved a better accuracy than using the best of individual data set, using all available data sets, and using selected features from conventional feature selection methods. Hill is also as good as the top 7% and 2% of the possible combinations of whole data sets, as verified by using the combination of whole data sets from an exhaustive search, on 5 functions of yeast and 3 types of protein sites, respectively. The limitation of Hill on larger data sets has motivated us to improve

Hill to handle more data sets. In subsection 5.4.1, we illustrate the "Greedy-Hill climbing method", Greedy-Hill, which is capable of handling more data sets.

### 5.4.1 The Greedy-Hill Climbing Algorithm

The main idea of whole data set feature selection is to treat a **whole data set** as one feature, and to iteratively choose the best combination of whole data sets from all available data sets using a machine learning algorithm, with a performance check at each cycle.

Greedy-Hill is a greedier version of Hill. Instead of the conservative one-set-per-cycle strategy of Hill, Greedy-Hill selects any data sets—there can be more than one—during each iteration, as long as their inclusion improves the classification performance. Let us describe Greedy-Hill now.

Let $D_{start}$ be a starting microarray data set that is to be analyzed with a classification learning algorithm $M$. Here, $M \in \{\text{C4.5, SVM, NBay, MLP}\}$. Typically, $D_{start}$ would be a new microarray data set generated by one's own laboratory.

Let $\Phi_{additional} = \{D_1, \ldots, D_n\}$ be $n$ additional microarray data sets conducted on the same set of genes as $D_{start}$. These additional data sets could be from different laboratories through various experimental studies.

Our goal is to exploit these additional data sets by searching for a $\Phi_{new} \subset \Phi_{additional}$ that can be combined with $D_{start}$ to give a better classification results by $M$.

We measure the classification performance of $M$ by the $S(M, 2)$ score. Here, let us denote $\mathcal{S}_M(\Psi)$ be the $S(M, 2)$ score of applying $M$ on the data sets in $\Psi \subseteq \{D_{start}\} \cup \Phi_{additional}$.

**Step 1:** Normalize the expression vectors in $D_{start}, D_1, \ldots, D_n$ so that they each have

mean 0 and standard deviation 1. Note that other appropriate normalization methods can also be used.

**Step 2:** Set $\Phi_{new}^{(0)} := D_{start}$.

**Step 3:** In the $i$-th iteration, $\Phi_{new}^{(i)}$ represents the subset of new data sets in $\Phi_{additional}$ that can be included to improve the classification performance of $M$. We determine $\Phi_{new}^{(i)}$ as follows:

- Set $\Phi_{new}^{(i)} := \emptyset$.
- For each $D_j \in \Phi_{additional}$, set $\Phi_{new}^{(i)} := \Phi_{new}^{(i)} \cup \{D_j\}$, if $\mathcal{S}_M, (\bigcup_{k=0}^{k \leq i} \Phi_{new}^{(k)} \bigcup \{D_j\}) > \mathcal{S}_M, (\bigcup_{k=0}^{k \leq i} \Phi_{new}^{(k)})$.

**Step 4:** After each iteration, we set $\Phi_{additional} := \Phi_{additional} - \Phi_{new}^{(i)}$.

We halt the process if either $\Phi_{additional}$ or $\Phi_{new}^{(i)}$ is $\emptyset$. Upon termination, we output $\bigcup_k \Phi_{new}^{(k)}$ as the desired combination of additional data sets from $\{D_1, ..., D_n\}$, that can be included with $D_{start}$ to give a better classification performance through $M$.

In Greedy-Hill, we liberalized the Hill algorithm to choose more data sets in each iteration through the possible candidate sets sequentially. Table 5.1 shows 'TCA' got the best combination of whole data sets as {Alp, Elu, Spo} in three cycles through Hill. When the same data is subjected to Greedy-Hill, the same best combination of whole data sets is selected in the first cycle itself. This means Greedy-Hill takes only one-cycle to complete the same task of Hill on this specific example.

### 5.4.2 Hill and Greedy-Hill

In this subsection we give a more pictorial explanation of our methods: Hill which chooses one data set per cycle, as in Figure 5.2; and Greedy-Hill which chooses multiple data sets per cycle, as in Figure 5.3. Later we show that Greedy-Hill is faster than Hill, and is useful when a large number of data sets are considered in the classification model.

To select the best combination of whole data sets for a class $f$, from $Dataset = \{D_{start}, D_1, D_2, \ldots, D_n\}$, the Hill algorithm is terminated at a cycle, when there is no further improvement of performance compared to using the starting set of that cycle or when there is no more data sets for forming new combinations.

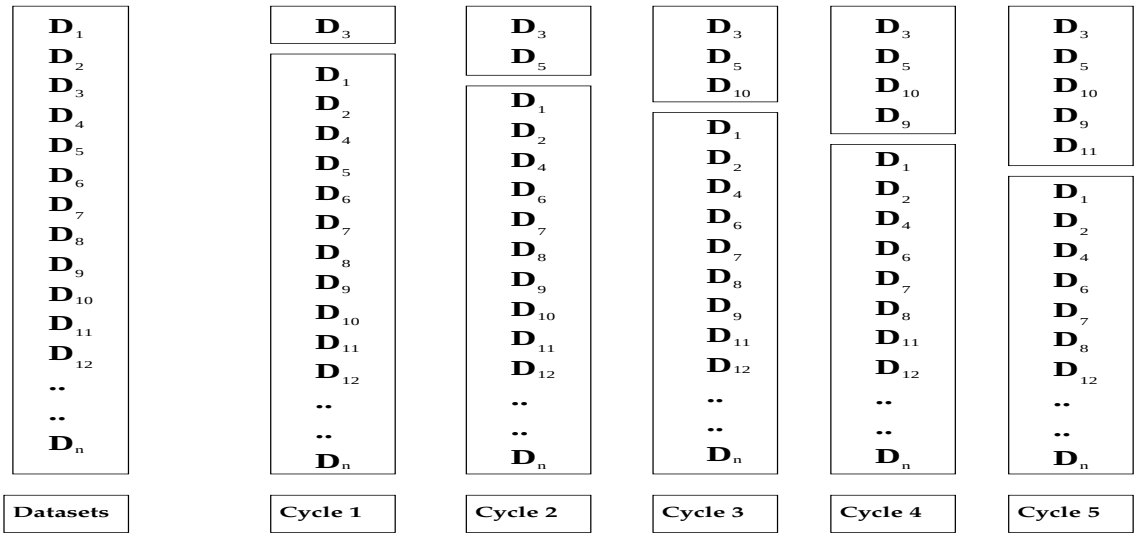


Figure 5.2: Selection of one data set per cycle by Hill.

We explain Figure 5.2 below:

Cycle 0: We have $Datasets = \{D_1, D_2, \ldots, D_n\}$, where $D_1, \ldots, D_n$ are the individual whole data sets and $f$ is a class.

Cycle 1: Hill chooses $D_3$ as the best performer for class $f$, from $Datasets$.

Cycle 2: The starting combination is now $\{D_3\}$. Hill evaluates the performance of combinations $\{D_3, D_1\}$, $\{D_3, D_2\}$, $\{D_3, D_4\}$, ..., $\{D_3, D_n\}$. The combination $\{D_3, D_5\}$ gives the best performance amongst these.

Cycle 3: The starting combination is now $\{D_3, D_5\}$. Similar to Cycle 2, Hill searches for further combinations that could achieve a higher performance than that of using only the starting combination. The combination $\{D_3, D_5, D_{10}\}$ yields the best performance in this cycle.

Cycle 4: The starting combination is now $\{D_3, D_5, D_{10}\}$. Similar to Cycle 2, the search for a better expanded combination is executed. The combination $\{D_3, D_5, D_{10}, D_9\}$ yields the best performance in this cycle.

Cycle 5: The starting combination is now $\{D_3, D_5, D_{10}, D_9\}$. Similar to Cycle 2, the search for a better expanded combination is executed. The combination of $\{D_3, D_5, D_{10}, D_9, D_{11}\}$ yields the best performance in this cycle.

Cycle 6: The starting combination is now $\{D_3, D_5, D_{10}, D_9, D_{11}\}$. Here, after evaluating all immediate expansions of this starting combination, Hill could not find one that would yield a better performance. Thus Hill stops and outputs $\{D_3, D_5, D_{10}, D_9, D_{11}\}$ as the best combination of whole data sets that gives the highest performance for the class $f$.

In contrast, the Greedy-Hill algorithm works by selecting multiple data sets in each cycle as per Figure 5.3:

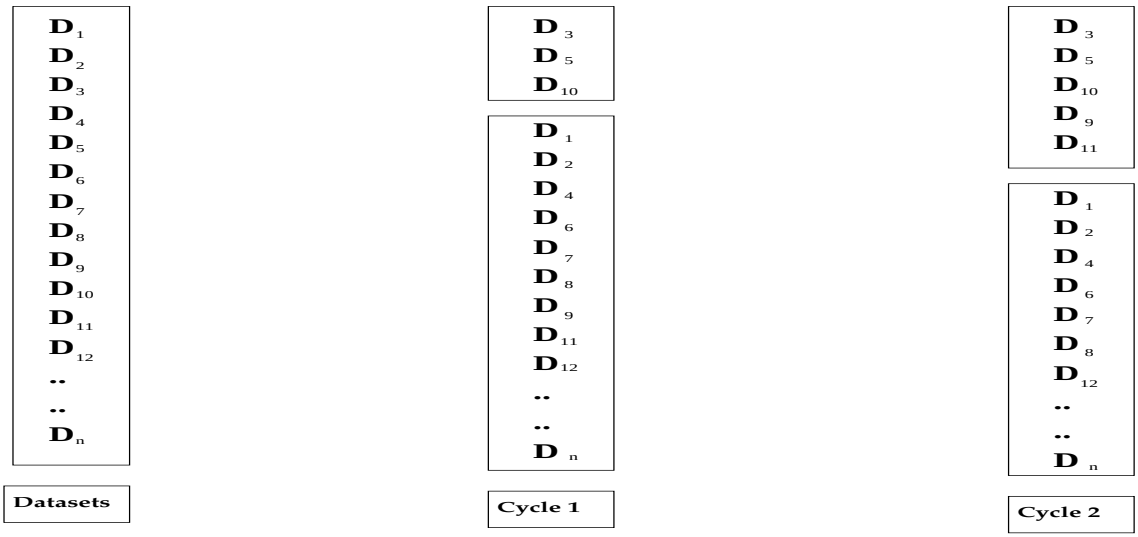


Figure 5.3: Selection of multiple data sets per cycle by Greedy-Hill.

Cycle 0: We have $Datasets = \{D_1, D_2, ..., D_n\}$, where $D_1, \ldots, D_n$ are the individual whole data sets.

Cycle 1: Greedy-Hill chooses $D_3$ from $Datasets$ for a class $f$, as it yields a high performance. The starting combination is now $\{D_3\}$ and the "pointing position" of unused sets in $Datasets$ is now at $D_3$.

Greedy-Hill expands the starting combination by trying the unused data sets from the current pointing position onwards. The combination $\{D_3, D_4\}$ does not produce a better performance. So Greedy-Hill advances the pointing position and finds that $\{D_3, D_5\}$ yields a better performance than using $D_3$. The starting combination set is now $\{D_3, D_5\}$ and the pointing position is now at $D_5$. Greedy-Hill continues to expand the starting combination by trying unused data sets from the current pointing position onwards. Advancing through pointing positions $D_6$, $D_7$, $D_8$, and $D_9$, the combinations $\{D_3, D_5, D_6\}$,

$\{D_3, D_5, D_7\}$, $\{D_3, D_5, D_8\}$, and $\{D_3, D_5, D_9\}$ do not yield a performance better than $\{D_3, D_5\}$. So Greedy-Hill continues to advance the pointing position to $D_{10}$. Greedy-Hill finds that $\{D_3, D_5, D_{10}\}$ yields a better performance than using $\{D_3, D_5\}$. So the starting combination is now set is changed to $\{D_3, D_5, D_{10}\}$. It continues to try to expand this combination by testing unused data sets from $D_{10}$ onwards. But these further combinations of $\{D_3, D_5, D_{10}\}$ with $D_{11}$, ..., $D_n$ do not produce an improved performance. This ends Cycle 1.

Cycle 2: The starting set is now $\{D_3, D_5, D_{10}\}$ and the pointing position is now back at $D_1$. Greedy-Hill tries to expand the combination $D_{start}$ by trying the unused data sets from $D_1$ onwards. The combinations $\{D_3, D_5, D_{10}, D_1\}$, $\{D_3, D_5, D_{10}, D_2\}$, ..., $\{D_3, D_5, D_{10}, D_8\}$ do not produce a better performance. But the combination $\{D_3, D_5, D_{10}, D_9\}$ yields a higher accuracy than using the current starting combination. So $D_9$ is added to the starting combination, and the pointing position is further advanced to test the next unused data set $D_{11}$. The combination $\{D_3, D_5, D_{10}, D_9, D_{11}\}$ yields a better performance, and $D_{11}$ is also added to the starting combination. This ends Cycle 2.

Cycle 3: As further combinations of $D_{start}$ with remaining data sets do not yield a better performance, the Greedy-Hill algorithm stops and returns $\{D_3, D_5, D_{10}, D_9, D_{11}\}$ as the best combination of whole data sets with the highest performance for the class $f$.

Greedy-Hill executes much faster than Hill, as it picks more useful data sets in each iteration than Hill. Greedy-Hill is useful in selecting the best combination of whole data sets on classification problems that consider a large number of data sets.

### 5.4.3 Using Combination Picked by Greedy-Hill on 5 Specific Functions of Yeast Genes

We have 8 data sets from Brown *et al.* (with genes annotated as per MIPS Catalogue (Version 1.3) of 25th June 2003 [36]), corresponding to 8 experimental conditions (different sets of features on same set of genes). This yields 255 ($= 2^8 - 1$) non-empty sets that involve the 8 data sets. We have a sufficiently small number of combinations of data sets here. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 5 specific functions $f \in$ {HIST, PROT, RESP, RIBO, TCA} using these 255 sets of data sets, and compare the performance $S(C_f(Greedy\text{-}Hill, m))$ using the combination of data subsets selected by Greedy-Hill. Results of this exhaustive study (abbreviated as EXH in the tables) are shown in Tables 5.21.

Table 5.21: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 5 functions of yeast through SVM.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| HIST | 245 | 10 | 0 | 96.08 | 3.92 | 0.00 |
| PROT | 211 | 5 | 39 | 82.75 | 1.96 | 15.29 |
| RESP | 203 | 52 | 0 | 79.61 | 20.39 | 0.00 |
| RIBO | 251 | 3 | 1 | 98.43 | 1.18 | 0.39 |
| TCA | 255 | 0 | 0 | 100.00 | 0.00 | 0.00 |

In this table, the second, third, and fourth columns show the number of sets that give poorer, equal, and better performance than $S(C_f(Greedy\text{-}Hill, m))$. The fifth, sixth, and seventh columns show the respective percentages with respect to the 255 total possible sets. The corresponding tables through NBay, C4.5, and MLP are tabulated (A.1, A.2, and A.3, respectively) in Appendix A. In fact, out of the 5100 combination of data sets through an exhaustive search over SVM, C4.5, NBay, and

MLP on the 5 specific cellular functions, 279 (=5%) of the possible combinations of whole data sets yield an accuracy equal to Greedy-Hill chosen data, 4664 (=92%) of the possible combinations of whole data sets yield lesser accuracy than Greedy-Hill chosen data, and 157 (=3%) of total combinations of data sets yield better accuracy than Greedy-Hill chosen data. This means that the combination chosen by Greedy-Hill is among the 8% of combinations that give the best performance, at least for the purpose of predicting the 5 specific functions of yeast genes.

When results of feature selection methods are compared with that of Greedy-Hill, we found that Greedy-Hill achieves better results in 11 out of 20 cases, equal results in 4 out of 20 cases, and lesser results in 5 out of 20 cases. This means Greedy-Hill is capable of achieving a better or equal performance by $S(M, 2)$ in 15 out of 20 cases.

### 5.4.4 Comparison of Hill vs Greedy-Hill on 5 Specific Functions of Yeast Genes

We have 8 data sets from Brown *et al.*, with genes annotated as per MIPS Catalogue (Version 1.3) of 25th June 2003 [36], corresponding to 8 experimental conditions (different sets of features on same set of genes). We already got the performances $S(C_f(Hill, m))$ and $S(C_f(Greedy\text{-}Hill, m))$ over 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 5 specific functions $f \in$ {HIST, PROT, RESP, RIBO, TCA}, using Hill and Greedy-Hill methods. Table 5.22 shows the number of functions $f$ that achieve a $S(M, 2)$ measure, based on the best combination of data subsets delivered by Hill, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10 to 13), the performance delivered by Greedy-Hill.

Table 5.22 shows that Hill and Greedy-Hill achieves similar prediction accuracies

Table 5.22: Total number and percentage of 5 functions of yeast for Hill>Greedy-Hill, Hill=Greedy-Hill, and Hill<Greedy-Hill through algorithms.

| | Grt. Greedy-Hill | | | | Eq. Greedy-Hill | | | | Les. Greedy-Hill | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 2 | 1 | 2 | 1 | 3 | 4 | 3 | 4 | 0 | 0 | 0 | 0 |
| % Functions | 40 | 20 | 40 | 20 | 60 | 80 | 60 | 80 | 0 | 0 | 0 | 0 |

for most of the functions through all learning algorithms (14 out of 20 cases). Table 5.23 shows that the average execution time of Greedy-Hill is always better than that of Hill, even for this small number of data subsets. This complexity advantage will be amplified when the number of data subsets is large.

Table 5.23: Comparison among Hill and Greedy-Hill, on average of performance by $S(M, 2)$ and time taken over 5 functions of yeast through algorithms.

| | Hill | | Greedy-Hill | |
|---|---|---|---|---|
| Algorithm | S(M,2) | Seconds | S(M,2) | Seconds |
| C4.5 | 53 | 76 | 50 | 51 |
| SVM | 56 | 303 | 55 | 225 |
| NBay | 49 | 64 | 48 | 56 |
| MLP | 58 | 6462 | 57 | 5137 |

The time taken by Hill is still quite large, and an analysis of the algorithm also shows that its time complexity grows as a square of the number of data subsets. So Hill is not suitable for problems where the number of data subsets is large. For a majority of gene functions (10 out of 20 cases), Hill achieves as good a performance as the best performance delivered by the exhaustive search. This is in contrast to conventional feature selection methods which matches the performance delivered by the exhaustive search only in 6 out of 20 cases. An important point to note also is that Hill is much faster than exhaustive search which checks all combinations of available data sets and selects the best one.

However, Greedy-Hill is much faster in selecting important data subsets than exhaustive search and Hill. In fact, a typical run of Greedy-Hill would take 1367 seconds, compared to 1726 seconds for Hill and 55308 seconds for exhaustive search. The average performance by $S(M, 2)$ on 5 functions of yeast, Greedy-Hill got 52.60, Hill yielded 53.80, and Exhaustive search yielded 56.55. Thus Hill should be used for classification problems where a small number of data subsets are considered, but Greedy-Hill should be used where a larger number of data subsets are encountered.

### 5.4.5 Using Combination Picked by Greedy-Hill on 3 Specific Types of Protein Sites

Recall that for annotated atoms there are 6 data sets. This yields 63 (= $2^6$ -1) non-empty sets. This is a sufficiently small number of sets. This enables us to consider the performance by $S(M, 2)$ of 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 3 types of protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE} using these 63 sets of sets, and compare the best performance on the same protein site with Greedy-Hill chosen data. Results of this exhaustive study are shown in Tables 5.24. In this table, the second, third, and fourth columns show the number of sets that give poorer, equal, and better performance than $S(C_s(\textit{Greedy-Hill}, m))$. The fifth, sixth, and seventh columns show the respective percentages with respect to the 63 total possible sets.

The corresponding tables through NBay, C4.5, and MLP are tabulated (B.1, B.2, and B.3, respectively) in Appendix B. In fact, out of the 756 combination of data sets through an exhaustive search over SVM, C4.5, NBay, and MLP on the 3 specific cellular functions, 20 (=3%) of the possible combinations of whole data sets yield an accuracy equal to Greedy-Hill chosen data, 696 (=92%) of the possible combinations

Table 5.24: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 3 types of protein sites through SVM.

| Types of Sites | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 47 | 3 | 13 | 74.603 | 4.762 | 20.635 |
| DISULFIDE | 45 | 2 | 16 | 71.429 | 3.175 | 25.397 |

of whole data sets yield lesser accuracy than Greedy-Hill chosen data, and 40 (=5%) of total combinations of data sets yield better accuracy than Greedy-Hill chosen data. This means that the combination chosen by Greedy-Hill is among the 8% of combinations that give the best performance, at least for the purpose of predicting the 3 specific types of protein sites.

When results of feature selection methods are compared with that of Greedy-Hill, we found that Greedy-Hill achieves better results in 6 out of 12 cases, equal results in 2 out of 12 cases, and lesser results in 4 out of 12 cases. This means Greedy-Hill is capable of achieving a better or equal performance by $S(M, 2)$ in 8 out of 12 cases.

### 5.4.6 Comparison of Hill vs Greedy-Hill on 3 Specific Types of Protein Sites

We have 6 data sets corresponding to 6 micro-environment properties. We already got the performances $S(D_s(Hill, m))$ and $S(D_s(Greedy\text{-}Hill, m))$ over 4 learning methods $m \in$ {C4.5, SVM, NBay, and MLP} for predicting the 3 types of protein sites $s \in$ {CALCIUM, SERINE, DISULFIDE}, using Hill and Greedy-Hill methods. Table 5.25 shows the number of protein sites $s$ that achieve a $S(M, 2)$ measure, based on the best combination of data subsets delivered by Hill, that is better than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is less than (Columns 10

to 13), the performance delivered by Greedy-Hill.

Table 5.25: Total number and percentage of 3 types of protein sites for Hill>Greedy-Hill, Hill=Greedy-Hill, and Hill<Greedy-Hill through algorithms.

| | Grt. Greedy-Hill | | | | Eq. Greedy-Hill | | | | Les. Greedy-Hill | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| Functions | 2 | 2 | 0 | 0 | 1 | 1 | 3 | 3 | 0 | 0 | 0 | 0 |
| % Functions | 40 | 40 | 0 | 0 | 20 | 20 | 60 | 60 | 0 | 0 | 0 | 0 |

Table 5.25 shows that Hill and Greedy-Hill achieves similar prediction accuracies for most of the protein sites through all learning algorithms (8 out of 12 cases). Table 5.26 shows that the average execution time of Greedy-Hill is similar to that of Hill, due to the small number of data subsets.

Table 5.26: Comparison among Hill and Greedy-Hill on average of performance by $S(M, 2)$ and time taken by over 3 types of protein sites through algorithms.

| | Hill | | Greedy-Hill | |
|---|---|---|---|---|
| Algorithm | S(M,2) | Seconds | S(M,2) | Seconds |
| C4.5 | 113 | 54 | 110 | 55 |
| NBay | 96 | 44 | 96 | 45 |
| SVM | 103 | 78 | 89 | 76 |
| MLP | 111 | 182 | 111 | 201 |

The time taken by Hill is still quite large, and an analysis of the algorithm also shows that its time complexity grows as a square of the number of data subsets. So Hill is not suitable for problems where the number of data subsets is large. For a majority of protein sites (9 out of 12 cases), Hill achieves as good a performance as the best performance delivered by the exhaustive search. This is in contrast to conventional feature selection methods which matches the performance delivered by the exhaustive search only in (1 out of 12 cases). An important point to note also is

that Hill (and also Greedy-Hill) is much faster than exhaustive search which checks all combinations of available data sets and selects the best one.

The average performance by $S(M, 2)$ on 3 types of protein types, Greedy-Hill got 101.00, Hill yielded 105.83, and Exhaustive search yielded 106.67. Thus Hill should be used for classification problems where a small number of data subsets are considered, but Greedy-Hill should be used where a larger number of data subsets are encountered.

## 5.5 Inferring Functions of *S. cerevisiae*

In Subsection 5.5.1, we present classification studies on functions of yeast genes using the combination of whole data sets chosen by Greedy-Hill.

In Subsection 5.5.2, we compare the performance by $S(M, 2)$ of using the combination of whole data sets chosen by Greedy-Hill to that of using the best of individual data sets, using all available data sets, and using the best of features selected by conventional feature selection methods.

### 5.5.1 The Study of 26 Functions of Yeast Genes Using Greedy-Hill Chosen Data

The focus of this chapter is to analyse the prediction accuracy of classification models using the combination of whole data sets chosen by Greedy-Hill—and show their merits and demerits. We have 57 data sets of gene expression experiments based on the 57 experimental conditions (different sets of features on same set of genes) (as detailed in Table 3.6)—$\mathcal{A}$ = {Alp1, Alp2, Alp3, Alp4, Ace, Des1, Des2, Des3, Des4, Des5, Des6, Des7, Haa, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15, Hea16, Hea17, Hea18, Hea19, Hea20,

Dby1, Dby2, Dby3, Dby4, Dby5, Dby6, Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8, Fch1, Fch2, Met, Hyd, Iro, Aft, Fit, Pho, Snf, Spo}.

Now, we explain how to build a classifier $C_f(\mathcal{A}, \textit{Greedy-Hill} + m)$ based on the combination of whole data sets chosen by Greedy-Hill for a machine learning method $m$ on the data set $\mathcal{A}$. Here, Greedy-Hill is applied on the training portion of the data to obtain a reduced data set, the machine learning method $m$ is then applied to the reduced data set to obtain a classification model which is then applied to the testing portion of the data set. Now, we apply Greedy-Hill on the data set $\mathcal{A}$ and build a classification model $C_f(\mathcal{A}, \textit{Greedy-Hill} + m)$, where $m \in$ {C4.5, SVM, NBay, MLP} and 26 functions $f \in$ {Aam, Nsm, Nuc, Pho, Ccm, Lim, Vit, Tca, Res, Fer, Mer, Ecr, Dna, Cyc, Tcp, Rsn, Rpr, Rmo, Pro, Rib, Tra, Trc, Ami, Pft, Pfs, Ptt, Prm, Apc, Deg, Tcs, Tfc, Trt, Int, Rdv, Str, Dtx, Hom, Csr, Tvp, Gro, Dea, Wal, Cyt, Nuc, Mit, Fun}.

We show below in Table 5.27 the results of the experiments just described, for 26 functions, using machine learning methods $m \in$ {C4.5, SVM, NBay, MLP}, for 26 specific functions $\in f$ annotated as per MIPS functional catalogue (Version 2.0) dated 19th March 2004, and using Greedy-Hill chosen data. The second through fifth columns are the performance $S(\textit{Greedy-Hill} + m, 2)$ through different learning algorithms $m \in$ {C4.5, SVM, NBay, MLP} from the WEKA package [65] at its default settings.

Table 5.27: Performance by $S(M, 2)$ on 26 functions of yeast based on Greedy-Hill selected data sets through algorithms.

| Function | Code | Genes | C4.5 | SVM | NBay | MLP |
|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | 20 | 16 | 3 | 7 |
| 11.04 | Rpr | 161 | 14 | 0 | 8 | 18 |
| 10.03 | Cyc | 149 | 7 | 11 | 2 | 9 |
| 20.09 | Trt | 145 | 0 | 2 | 4 | 2 |
| 12.01 | Rib | 138 | 221 | 239 | 208 | 249 |
| 1.01 | Aam | 103 | 65 | 59 | 11 | 74 |
| 1.06 | Lim | 99 | 18 | 0 | 0 | 20 |
| 10.01 | Dna | 99 | 17 | 3 | 4 | 20 |
| 1.05 | Ccm | 82 | 2 | 2 | 0 | 4 |
| 1.03 | Nuc | 81 | 13 | 14 | 0 | 22 |
| 14.13 | Deg | 77 | 32 | 0 | 7 | 38 |
| 32.01 | Str | 58 | 3 | 7 | 1 | 8 |
| 1.07 | Vit | 54 | 0 | 0 | 0 | 0 |
| 14.07 | Prm | 48 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | 0 | 0 | 0 | 13 |
| 12.04 | Tra | 42 | 0 | 0 | 1 | 8 |
| 11 | Tcp | 39 | 0 | 0 | 0 | 1 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | 15 | 15 | 0 | 16 |
| 20.03 | Tfc | 32 | 0 | 0 | 0 | 0 |
| 42.01 | Wal | 32 | 0 | 0 | 0 | 0 |
| 12.10 | Ami | 31 | 13 | 5 | 0 | 21 |
| 43.01 | Fun | 31 | 0 | 0 | 0 | 1 |
| 2.13 | Res | 29 | 12 | 11 | 4 | 25 |
| 14.01 | Pfs | 29 | 0 | 0 | 0 | 4 |
| 32.07 | Dtx | 27 | 0 | 8 | 0 | 4 |

### 5.5.2 Comparison of Greedy-Hill Chosen Data to Best Individual Data Sets, All Available Data Sets, and Selected Features

In the previous subsection we showed the accuracy of inferring if a gene has one of the 26 specific functions (as detailed in Section 5.5.1) based on a feature vector derived from the gene expression profile of that gene by Greedy-Hill. In this subsection, we compare Greedy-Hill results with that of using the best of individual data sets, all available data sets, and features selected by conventional feature selection methods.

We have 57 data sets—$\mathcal{A}$ = {Alp1, Alp2, Alp3, Alp4, Ace, Des1, Des2, Des3, Des4, Des5, Des6, Des7, Haa, Hea1, Hea2, Hea3, Hea4, Hea5, Hea6, Hea7, Hea8, Hea9, Hea10, Hea11, Hea12, Hea13, Hea14, Hea15, Hea16, Hea17, Hea18, Hea19, Hea20, Dby1, Dby2, Dby3, Dby4, Dby5, Dby6, Cal1, Cal2, Cal3, Cal4, Cal5, Cal6, Cal7, Cal8, Fch1, Fch2, Met, Hyd, Iro, Aft, Fit, Pho, Snf, Spo}. We got the performance

by $S(M,2)$ for a function $f$, $S(M,2) = \max_{C \in \mathcal{A}} C_f(C,m)$, using the best of individual data sets and various learning methods $m \in \{$C4.5, SVM, NBay, MLP$\}$, for a function $f \in \{$Aam, Nsm, Nuc, Pho, Ccm, Lim, Vit, Tca, Res, Fer, Mer, Ecr, Dna, Cyc, Tcp, Rsn, Rpr, Rmo, Pro, Rib, Tra, Trc, Ami, Pft, Pfs, Ptt, Prm, Apc, Deg, Tcs, Tfc, Trt, Int, Rdv, Str, Dtx, Hom, Csr, Tvp, Gro, Dea, Wal, Cyt, Nuc, Mit, Fun$\}$, annotated as per the MIPS Catalogue (Version 2.0) of 19th March 2004.

We already got the performance $S(Greedy\text{-}Hill+m,2)$ from $C_f(\mathcal{A}, Greedy\text{-}Hill+m,2)$, for a function $f$. Table 5.28 shows the number of functions (26 functions), $f$ that achieves a $S(M,2)$ measure, based on the performance of $S(Greedy\text{-}Hill+m,2)$ for the combination of whole data sets chosen of Greedy-Hill that is less than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is better than (Columns 10 to 13), the best of individual data sets, for machine learning method $m$. Here, we use implementations in the WEKA package [65] with default settings. It is clear from the table that, for most protein functions, Greedy-Hill gives a higher performance by $S(M,2)$ than using the best of individual data sets.

Table 5.28: Number and percentage for Greedy-Hill<BI, Greedy-Hill=BI, and Greedy-Hill>BI on 26 functions of yeast through algorithms.

| | Les. *BI* | | | | Eq. *BI* | | | | Grt. *BI* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Greedy-Hill-BI | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| 26 Functions | 0 | 0 | 0 | 0 | 12 | 15 | 16 | 8 | 14 | 11 | 10 | 18 |
| % Functions | 0 | 0 | 0 | 0 | 46 | 58 | 62 | 31 | 54 | 42 | 38 | 69 |

Table 5.28 further shows that Greedy-Hill climbing yields equal performance by $S(M,2)$ in 49% of the functions and greater in 51% to the Best Individual data sets, over 4 methods.

We calculate the performance by $S(M,2)$ for a function $f$, from $C_f(ALL,m)$,

where $ALL$ is the data set derived by combining or merging 57 data sets annotated as per the MIPS Catalogue (Version 2.0) of 19th March 2004 (as detailed in Section 5.5.1).

We already got the performance $S(Greedy\text{-}Hill+m, 2)$ from $C_f(\mathcal{A}, Greedy\text{-}Hill+m)$, for a function $f$. Table 5.29 shows the number of functions (26 functions), $f$ that achieves a $S(M, 2)$ measure, based on the performance of $S(Greedy\text{-}Hill + m, 2)$ for the combination of whole data sets chosen of Greedy-Hill that is less than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is better than (Columns 10 to 13), all the available data sets for machine learning method $m$. Here, we use the implementations in the WEKA package [65] with default settings. It is clear from the table that, for most protein functions, Greedy-Hill gives a higher performance by $S(M, 2)$ than using all available data sets.

Table 5.29: Number and percentage for Greedy-Hill<ALL, Greedy-Hill=ALL, and Greedy-Hill>ALL on 26 functions of yeast through algorithms.

| | Les. *ALL* | | | | Eq. *ALL* | | | | Grt. *ALL* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Greedy-Hill-ALL | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| 26 Functions | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 26 | 25 | 26 | 24 |
| % Functions | 0 | 4 | 0 | 4 | 0 | 0 | 0 | 4 | 100 | 96 | 100 | 92 |

Table 5.29 further shows that Greedy-Hill yields lesser in 2% of the functions, equal in 1%, and greater in 97% to ALL data sets, over 4 methods.

We also got the performance $S(FS + m, 2)$ from $C_f(ALL, FS + m)$, for feature selection methods $FS \in \{$CFS, Chi, Info$\}$, learning methods $m \in \{$C4.5, SVM, NBay, MLP$\}$, and functions $f \in \{$Aam, Nsm, Nuc, Pho, Ccm, Lim, Vit, Tca, Res, Fer, Mer, Ecr, Dna, Cyc, Tcp, Rsn, Rpr, Rmo, Pro, Rib, Tra, Trc, Ami, Pft, Pfs, Ptt, Prm, Apc, Deg, Tcs, Tfc, Trt, Int, Rdv, Str, Dtx, Hom, Csr, Tvp, Gro, Dea, Wal, Cyt,

Nuc, Mit, Fun}, annotated as per the MIPS Catalogue (Version 2.0) of 19th March 2004.

We already got the performance $S(Greedy\text{-}Hill+m, 2)$ from $C_f(\mathcal{A}, Greedy\text{-}Hill+m)$, for a function $f$. Table 5.30 shows the number of functions (26 functions), $f$ that achieves a $S(M, 2)$ measure, based on the performance of $S(Greedy\text{-}Hill+m, 2)$ for the combination of whole data sets chosen of Greedy-Hill that is less than (Columns 2 through 5), or that is equal to (Columns 6 through 9), or that is better than (Columns 10 to 13), the features selected by conventional feature selection methods for machine learning method $m$. Here, we use the implementations in the WEKA package [65] with default settings. It is clear from the table that, for most protein functions, Greedy-Hill gives a higher performance by $S(M, 2)$ than using features selected by conventional feature selection methods.

Table 5.30: Number and percentage for Greedy-Hill<FS, Greedy-Hill=FS, and Greedy-Hill>FS on 26 functions of yeast through algorithms.

| | Les. $FS$ | | | | Eq. $FS$ | | | | Grt. $FS$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Greedy-Hill-FS | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP | C4.5 | SVM | NBay | MLP |
| 26 Functions | 1 | 2 | 0 | 0 | 8 | 9 | 1 | 3 | 17 | 15 | 25 | 23 |
| % Functions | 4 | 8 | 0 | 0 | 31 | 35 | 4 | 12 | 65 | 58 | 96 | 88 |

### 5.5.3 Using Greedy-Hill Chosen Data Improves Prediction Accuracy on 26 Functions of Yeast Genes

In this subsection we summarizes the performance by $S(M, 2)$ on different approaches of 26 function of yeast. Table 5.31 shows the performance by $S(M, 2)$ through SVM using all available data sets, using the best of individual data sets, using selected features by conventional feature selection methods, and using the combination of

whole data sets chosen by Greedy-Hill. The rows are the 26 functions (as detailed in Section 5.5.1) shown in column 1 as catalogue number and column 2 as function code. Column 3 shows the number of genes for each function. Column 4 shows the performance $S(ALL{+}SVM, 2)$ for a function $f$ using all available data sets. Column 5 shows the performance $S(M, 2)$ for a function $f$ using the best of individual data sets. Column 6 shows the performance $S(Hill + SVM, 2)$ for a protein function $f$ using the combination of whole data sets chosen by Hill. Column 7 shows the performance $S(Greedy\text{-}Hill + SVM, 2)$ for a protein function $f$ using the combination of whole data sets chosen by Greedy-Hill. Column 8 shows the performance $S(CFS{+}SVM, 2)$ for a function $f$ using selected features by Correlation-based feature selection method. Column 9 shows the performance $S(Chi + SVM, 2)$ for a function $f$ using selected features by Chi-square feature selection method. Column 10 shows the performance $S(Info{+}SVM, 2)$ for a function $f$ using selected features by information-gain feature selection method.

Results through other learning algorithms—C4.5, NBay, and MLP—are tabulated (Table C.1, Table C.3, and Table C.2) for 26 functions in Appendix C.

We could not tabulate outcome on 20 functions by algorithms—C4.5, NBay, and MLP, due to space constraint.

Finally, we calculate the average on performance by $S(M, 2)$—using all available data sets (ALL), using the best of individual data sets (BI), using Greedy-Hill, using selected features from Correlation-based Feature Selection method (CFS), using selected features from Chi-square feature selection method (Chi), and using selected features from information-gain feature selection method (Info)—over 4 machine learning

Table 5.31: Performance by $S(M, 2)$ of 26 functions of yeast through SVM, using all available data sets, using the best of individual data sets, using the best combination of whole data sets chosen by Hill and Greedy-Hill, and using selected features from feature selection methods CFS, Chi, Info.

| 26 Functions | Code | Genes | S(ALL) | S(Best-Ind) | S(Hill) | S(Greedy-Hill) | S(CFS) | S(Chi) | S(Info) |
|---|---|---|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | -149 | 16 | 16 | 16 | -10 | -58 | -58 |
| 11.04 | Rpr | 161 | -140 | 0 | 0 | 0 | -28 | -69 | -68 |
| 10.03 | Cyc | 149 | -137 | 0 | 2 | 11 | -8 | -57 | -58 |
| 20.09 | Trt | 145 | -119 | 0 | 2 | 2 | -1 | -67 | -67 |
| 12.01 | Rib | 138 | 228 | 214 | 237 | 239 | 223 | 222 | 221 |
| 1.01 | Aam | 103 | 46 | 11 | 37 | 59 | 56 | 25 | 25 |
| 1.06 | Lim | 99 | -63 | 0 | 0 | 0 | 0 | -5 | -5 |
| 10.01 | Dna | 99 | -83 | 0 | 6 | 3 | -1 | -48 | -48 |
| 1.05 | Ccm | 82 | -63 | 2 | 2 | 2 | -6 | -47 | -47 |
| 1.03 | Nuc | 81 | -27 | 0 | 10 | 14 | 3 | 8 | 8 |
| 14.13 | Deg | 77 | -15 | 0 | 0 | 0 | 11 | -29 | -26 |
| 32.01 | Str | 58 | -8 | 4 | 8 | 7 | -1 | -23 | -23 |
| 1.07 | Vit | 54 | -34 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.07 | Prm | 48 | -61 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | -20 | 0 | 0 | 0 | 0 | -4 | -4 |
| 12.04 | Tra | 42 | -24 | 0 | 0 | 0 | -8 | -54 | -54 |
| 11 | Tcp | 39 | -34 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.04 | Ptt | 37 | -32 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | -5 | 6 | 11 | 15 | 7 | 11 | 11 |
| 20.03 | Tfc | 32 | -23 | 0 | 0 | 0 | 0 | 0 | 0 |
| 42.01 | Wal | 32 | -28 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12.10 | Ami | 31 | 8 | 0 | 1 | 5 | 0 | -7 | -7 |
| 43.01 | Fun | 31 | -13 | 0 | 0 | 0 | 0 | -2 | -2 |
| 2.13 | Res | 29 | 3 | 0 | 2 | 11 | 8 | -1 | -1 |
| 14.01 | Pfs | 29 | -34 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32.07 | Dtx | 27 | -8 | 2 | 4 | 8 | 0 | 0 | 0 |

Table 5.32: Performance by $S(M, 2)$ of 20 functions of yeast through SVM, using all available data sets, using the best of individual data sets, using the best combination of whole data sets chosen by Hill and Greedy-Hill, and using selected features from feature selection methods CFS, Chi, Info.

| 20 Functions | Code | Genes | S(ALL) | S(Best-Ind) | S(Hill) | S(Greedy-Hill) | S(CFS) | S(Chi) | S(Info) |
|---|---|---|---|---|---|---|---|---|---|
| 11.06 | Rmo | 20 | -8 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34.01 | Hom | 20 | 1 | 0 | 3 | 3 | 0 | 0 | 0 |
| 14.10 | Apc | 17 | -13 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30.01 | Int | 16 | -11 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1.02 | Nsm | 13 | 0 | 0 | 4 | 4 | 0 | 0 | 0 |
| 12.07 | Trc | 12 | -3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1.04 | Pho | 11 | -2 | 0 | 3 | 2 | -2 | -9 | -9 |
| 40.01 | Gro | 10 | -3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 42.16 | Mit | 9 | -14 | 0 | 0 | 0 | 0 | 0 | 0 |
| 38 | Tvp | 8 | 1 | 0 | 4 | 5 | 0 | 0 | 0 |
| 2.16 | Fer | 7 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2.45 | Ecr | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | Pro | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2.10 | Tca | 6 | -3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32 | Rdv | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 42.10 | Nuc | 5 | -1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2.19 | Mer | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | Pft | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 40.10 | Dea | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 42.04 | Cyt | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

method. Results from these approaches are—ALL:−32.12, BI:9.8, Hill:13, Greedy-Hill:15.1, CFS:9.4, Chi:−7.88, Info:−7.81. Thus, Greedy-Hill:15.1, achieves the best performance by $S(M, 2)$ for the purpose of predicting 26 functions of yeast. This tells us that we have achieved a better average performance by Greedy-Hill and Hill methods than other conventional methods.

## 5.6 Conclusion on Use of Hill Climbing Methods

Biologists need to have a better model with useful experiments for functional studies. In this chapter, we formulated a hypothesis that using the combination of whole data sets chosen by "Hill climbing methods" yields a better performance than other conventional methods. We demonstrated this on 3 specific examples:

1. 5 functions of yeast were studied, where the combination of whole data sets chosen by Hill was compared with the best of individual data sets, all available data sets, selected features from feature selection methods, and the best combination of whole data sets through exhaustive search. We have also applied Greedy-Hill on 5 functions of yeast and compared the results with that of exhaustive search and conventional feature selection methods.

2. 3 types of protein sites were studied with micro-environment properties surrounding protein site, where the combination of whole sets of micro-environment properties chosen by Hill was compared with the best of individual sets of micro-environment properties, all available sets of micro-environment properties, selected sets of micro-environment properties from conventional feature selection methods, and the best combination of whole sets of micro-environment properties through exhaustive search. We have also applied Greedy-Hill on 3 types

of protein sites and compared the results with that of exhaustive search and feature selection methods.

3. 26 cellular functions of yeast from MIPs functional annotation comprising 57 data sets were studied, where the combination of whole data sets chosen by Greedy-Hill was compared with the best of individual data sets, all available data sets, and selected features from conventional feature selection methods.

The point to note is that previous researchers (Brown *et al.*, Mateos *et al.*, Bagley *et al.* and Wei *et al.*) used all available data sets together as one single combined data set in their classification studies. They did not show whether using all available data sets would consistently lead to better performance than using a judiciously chosen smaller combination of whole data sets. They also did not investigate the issue of the optimal choice of combinations of whole data sets.

We demonstrated that using the combinations of whole data sets chosen by Hill and Greedy-Hill lead to best performance, compared to using the best of individual data sets or using all available data sets or using selected features from conventional feature selection methods. Also Hill is as good as the top 7% and 2% of the possible combinations of whole data sets, as verified by an exhaustive search on 5 functions of yeast and 3 types of protein sites respectively. The study of 5 functions of yeast by Hill is already reported in the conference GIW2003 [51].

The average performance delivered by the combination of data subsets chosen by Hill and Greedy-Hill are mostly comparable to the best solutions by exhaustive search. While Hill can provide a slightly better solution than Greedy-Hill, the later can reach a solution in much lesser time and can thus be used when there is a large number of extra data sets to be incorporated. Greedy-Hill also achieve better accuracy for majority

of cases than the conventional feature selection methods. As microarray technologies become routinely applied in genome laboratories for studying gene expression, it is expected that an increasing number of data sets will become available. Efficient algorithms such as Greedy-Hill will allow us to fully exploit these additional data sources to achieve better biological data mining results.

## 5.7 Differences in Treatment of Data

In this section, we discuss some differences between our results and that of Eisen *et al.* and Brown *et al.* on 5 functions of yeast, the differences in the treatment of data between Wei *et al.* and us on 3 types of protein sites, and data sets and functional differences on 26 functions of yeast.

### 5.7.1 5 Functions of Yeast Genes

To the best of our knowledge, the only significant previous work on inferring these 5 specific cellular functions, based on these 6 data sets, are that of Brown *et al.* [5] and Mateos *et al.* [34]. Both of them obtained classification accuracy similar to ours, though the numbers cannot be directly compared due to certain differences in the treatment of the data sets between them and us. Nevertheless, both Brown *et al.* and Mateos *et al.* used all available data sets, and did not consider the issue of using single data sets vs using multiple data sets.

In our study, we use the microarray data sets from Eisen *et al.* [10], available at the URL {http://rana.lbl.gov/EisenData.htm}. These data sets comprise 80 time points of 6 experimental assays on 6221 *S. cerevisiae* genes. We annotated the genes for the 5 functions using the MIPS Catalogue (Version 1.3) of 25th June 2003. This

gives 2550 genes with known functions, including 22 genes being annotated as TCA genes, 24 as RESP genes, 129 as RIBO genes, 33 as PROT genes, and 11 as HIST genes. In contrast, Brown *et al.* [5] used 64 time points from Eisen *et al.*, and added 14 time points of their own. Furthermore, they annotated the genes using the MIPS Catalogue of 8th November 2001. On the other hand, Mateos *et al.* [34] used 79 time points from Eisen *et al.* [10]. They also annotated the genes using the MIPS Catalogue of 8th November 2001, giving 2467 genes, including 17 genes being annotated as TCA genes, 27 as RESP genes, 121 as RIBO genes, 35 as PROT genes, and 11 as HIST genes.

In our study, we use the WEKA software package [65], as this package provides a large number of machine learning methods with a convenient interface. In contrast, Brown *et al.* [5] used the GIST software package which provides the SVM method only. Furthermore, Brown *et al.* tuned the SVM kernel function to handle less samples and to give more weight for positive samples. On the other hand, Mateos *et al.* used a multilayer perceptron, with one input layer comprising 79 units (one per experimental condition), one hidden layer comprising eight units, and one output layer with five units (one for each function).

Hence, the data sets, functional annotations, and classification methods are not the same among the studies conducted on yeast by Brown *et al.*, Mateos *et al.*, and us. Keeping these differences in mind, for the sake of completeness, we summarize in Table 5.33 the classification performance by $S(M, 2)$ on 5 functions of yeast from Brown *et al.*, Mateos *et al.* and our study. Both of them obtained classification accuracy similar to ours, though the numbers cannot be directly compared due to certain differences in the treatment of the data sets between them and us. Nevertheless, both

Brown *et al.* and Mateos *et al.* used all available data sets, and did not consider the issue of using single data sets vs using multiple data sets.

Table 5.33: Comparison of performances by $S(M, 2)$ of Brown and Mateos on all data sets, and ours on combination of data subsets chosen by Hill and best of exhaustive search on 5 functions of yeast.

| Function | Genes: Brown/ Mateos | Genes: In our Study | Performance by $S(M,2)$ through SVM | | | Performance by $S(M,2)$ through MLP | | |
|---|---|---|---|---|---|---|---|---|
| | | | S(Brown) | S(Sub-set) By Hill | S(Sub-set) By Exh | S(Mateos) | S(Sub-set) By Hill | S(Sub-set) By Exh |
| TCA | 17 | 22 | 12 | 8 | 7 | 3 | 4 | 12 |
| RESP | 27 | 25 | 38 | 0 | 0 | 28 | 0 | 2 |
| RIBO | 121 | 125 | 229 | 232 | 233 | 228 | 230 | 235 |
| PROT | 35 | 33 | 51 | 24 | 38 | 41 | 39 | 40 |
| HIST | 11 | 11 | 18 | 16 | 16 | 18 | 16 | 16 |

### 5.7.2 3 Types of Protein Sites

To the best of our knowledge, Bagley *et al.* [57] and Wei *et al.* [31] have one of the best results in classifying calcium binding sites, serine protease active sites, and disulfide bridges. Their reported accuracies are similar to ours, though the numbers cannot be directly compared due to differences in data sets used—we are unable to obtain their data sets. Nevertheless, both Bagley *et al.* and Wei *et al.* used all available sets of micro-environment properties data, and did not consider the issue of using single data sets vs using multiple data sets.

In our study, we use 174 atoms comprising—94 atoms for CALCIUM, 43 atoms for SERINE, and 37 atoms for DISULFIDE. We take atoms belonging to any protein site $s \in \{\text{CALCIUM, SERINE, DISULFIDE}\}$ as belonging to positive class and atoms belonging to other protein sites as belonging to negative class, in our classification model. In contrast, Wei *et al.* [31] use 16 CALCIUM sites and 100 randomly chosen non-binding sites in their classification model. Wei *et al.* [29] use 90 disulfide-binding

cysteins from 16 non-redundant proteins and 48 free cysteins—as non-sites—from 19 non-redundant proteins, in their study.

In our study, we use the WEKA package [65] which covers a large number of data mining algorithms with friendly interface. In contrast, Wei *et al.* [31] use the "FEATURE" package developed in-house at their lab.

The number of atoms, proteins, and classification algorithms are not the same on 3 types of protein sites among the studies conducted by Wei et al and us. Keeping these differences in mind, for the sake of completeness, we summarize in Table 5.34 the classification results from different studies. In this table we give prediction accuracy in terms of "sensitivity" and "specificity" which are calculated as—$Sensitivity = TP/(TP + FN)$ and $Specificity = TN/(TN + FP)$, where TP is the number of True positives, FN is the number of False negatives, TN is the number of True negatives, and FP is the number of False positives that are achieved after using an machine learning algorithm $m$. The "sensitivity" and "specificity" from Wei et al [31] and Wei et al [29] through Bayesian are given in columns 2-3, respectively, in the format "conducted by:sensitivity" and "conducted by:specificity". "Sensitivity" and "Specificity" that we obtained using the best of individual data set, using ALL data sets and selected combinations of data sets by Hill in column 4-5, columns 6-7 and columns 8-9, respectively.

### 5.7.3 26 Functions of Yeast Genes

16 data sets are taken in our study. 6 data sets are partitioned into 47 data sets. A total of 57 data sets based on experimental conditions are used on 2114 genes with 26 second level functional annotations from the MIPS Catalogue (Version 2.0) dated 19 March 2004. In contrast, Mateos *et al.* [34] use 6 data sets from Spellman *et al.*,

Table 5.34: SN:Sensitivity and SP:Specificity on ALL data from Wei et al (Wei1 [31], Wei2 [29]) through Bayesian and ours on BI, ALL, and Hill through MLP.

| Types of protein sites | Earlier studies | | Our Research Study | | | | | |
|---|---|---|---|---|---|---|---|---|
| | $ALL(Bayesian)$ | | $BI(MLP)$ | | $ALL(MLP)$ | | $Hill(MLP)$ | |
| | SN | SP | SN | SP | SN | SP | SN | SP |
| CALCIUM | Wei1:0.91 | Wei1:1.00 | 0.92 | 0.99 | 0.99 | 0.90 | 0.99 | 0.99 |
| SERINE | — | — | 1.00 | 0.95 | 0.95 | 0.91 | 1.00 | 0.99 |
| DISULFIDE | Wei2:88.9 | Wei2:97.9 | 0.86 | 1.00 | 0.97 | 0.97 | 0.92 | 0.98 |

and functions from MIPS Catalogue of 8 November 2001.

## 5.8 Issues to Further Validate Progressive Data Mining

We have illustrated in this chapter, how Progressive Data Mining on the selected combination of whole data sets through Hill and Greedy-Hill methods yield better performances than conventional methods. Some researchers advised to validate our results on multiple evaluation metrics, using committee of features, using committee method, and use of statistical sampling in selecting negative samples. In this section we address these issues.

### 5.8.1 Multiple Evaluation Metrics

In this report we tabulated performances by different approaches evaluated by Cost Saving function:S(M,2) (this is one of the 8 evaluation metrics) on all 3 research problems. Here, we summarize the performances through 8 evaluation metrics—S(M,2), Sensitivity, Precision, FM, Specificity, Accuracy, Rt F N, and Rt F P (as detailed in Subsection 3.4).

In Tables 5.35, 5.36, 5.37, 5.38 we show average performances over 5 functions of yeast, 3 types of protein sites, 26 functions, and 20 functions of yeast for all available data sets (ALL), the best of individual data sets (BI), selected features from Correlation-based Feature Selection method (CFS), selected features from Chi-square feature selection method (Chi), selected features from information-gain feature selection method (Info), Hill, Greedy-Hill, and exhaustive search (EXH) through SVM.

Table 5.35: Average performances over 5 functions of yeast by multiple evaluation metrics through SVM.

| DATA | S(M,2) | Sensitivity | Precision | FM | Specificty | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|
| BI | 35 | 0.296 | 0.379 | 0.332 | 0.999 | 0.989 | 0.700 | 0.001 |
| ALL | 48 | 0.587 | 0.363 | 0.354 | 0.994 | 0.989 | 0.409 | 0.006 |
| CFS | 47 | 0.342 | 0.472 | 0.385 | 0.998 | 0.991 | 0.651 | 0.002 |
| Chi | 55 | 0.564 | 0.567 | 0.563 | 0.997 | 0.992 | 0.433 | 0.003 |
| Info | 55 | 0.564 | 0.567 | 0.563 | 0.997 | 0.992 | 0.433 | 0.003 |
| Hill | 56 | 0.596 | 0.578 | 0.580 | 0.996 | 0.992 | 0.401 | 0.004 |
| Greedy-Hill | 55 | 0.584 | 0.574 | 0.573 | 0.996 | 0.992 | 0.413 | 0.004 |
| EXH | 58 | 0.522 | 0.646 | 0.566 | 0.999 | 0.994 | 0.475 | 0.001 |

Table 5.36: Average performances over 3 types of protein sites by multiple evaluation metrics through SVM.

| DATA | S(M,2) | Sensitivity | Precision | FM | Specificty | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|
| BI | 49 | 0.348 | 0.850 | 0.432 | 0.949 | 0.796 | 0.652 | 0.050 |
| ALL | 93 | 0.876 | 0.865 | 0.869 | 0.927 | 0.913 | 0.124 | 0.073 |
| CFS | 68 | 0.506 | 0.909 | 0.626 | 0.977 | 0.856 | 0.494 | 0.024 |
| Chi | 98 | 0.901 | 0.883 | 0.892 | 0.940 | 0.933 | 0.099 | 0.064 |
| Info | 98 | 0.901 | 0.883 | 0.892 | 0.940 | 0.933 | 0.099 | 0.064 |
| Hill | 103 | 0.919 | 0.943 | 0.927 | 0.975 | 0.956 | 0.081 | 0.024 |
| Greedy-Hill | 89 | 0.796 | 0.854 | 0.822 | 0.952 | 0.910 | 0.196 | 0.047 |
| EXH | 103 | 0.919 | 0.943 | 0.927 | 0.975 | 0.956 | 0.081 | 0.025 |

Table A.4 shows averages by C4.5, NBay, and MLP in Appendix A for 5 specific functions of yeast. Table B.4 shows averages by C4.5, NBay, and MLP in Appendix B for 3 types of protein sites. Table C.4 show averages by C4.5, NBay, and MLP in

Table 5.37: Average performances over 26 functions of yeast by multiple evaluation metrics through SVM.

| DATA | S(M,2) | Sensitivity | Precision | FM | Specificty | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|
| BI | 10 | 0.043 | 0.236 | 0.055 | 1.000 | 0.967 | 0.957 | 0.000 |
| ALL | -32 | 0.195 | 0.196 | 0.191 | 0.966 | 0.941 | 0.805 | 0.034 |
| CFS | 10 | 0.089 | 0.238 | 0.116 | 0.995 | 0.965 | 0.898 | 0.005 |
| Chi | -8 | 0.157 | 0.198 | 0.161 | 0.980 | 0.954 | 0.830 | 0.021 |
| Info | -8 | 0.158 | 0.198 | 0.161 | 0.980 | 0.954 | 0.829 | 0.021 |
| Hill | 13 | 0.067 | 0.393 | 0.092 | 0.999 | 0.968 | 0.933 | 0.000 |
| Greedy-Hill | 15 | 0.109 | 0.190 | 0.118 | 0.994 | 0.964 | 0.891 | 0.011 |

Table 5.38: Average performances over 20 functions of yeast by multiple evaluation metrics through SVM.

| DATA | SM | Sensitivity | Precision | FM | Specificty | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|
| BI | 0 | 0.000 | 0.000 | 0.000 | 1.000 | 0.996 | 1.000 | 0.000 |
| ALL | -3 | 0.027 | 0.070 | 0.037 | 0.998 | 0.994 | 0.973 | 0.002 |
| CFS | 0 | 0.000 | 0.000 | 0.000 | 0.750 | 0.746 | 0.490 | 0.000 |
| Chi | 0 | 0.007 | 0.004 | 0.005 | 0.750 | 0.746 | 0.486 | 0.000 |
| Info | 0 | 0.007 | 0.004 | 0.005 | 0.750 | 0.746 | 0.486 | 0.000 |
| Hill | 1 | 0.034 | 0.137 | 0.052 | 1.000 | 0.996 | 0.966 | 0.000 |
| Greedy-Hill | 1 | 0.024 | 0.020 | 0.020 | 0.998 | 0.994 | 0.976 | 0.000 |

Appendix C on 26 functions of yeast. The above tables demonstrate that Greedy-Hill and Hill scheme can achieve better performances than other conventional methods.

### 5.8.2 Committee of Features

In previous subsection, we showed that the performances through 8 evaluation metrics convinces that Hill and Greedy-Hill methods achieve better performances than conventional methods. In this subsection we use only $S(M, 2)$ measure to validate performances using committee of features.

**Cycle of Features :** Conventional feature selection method, $FS \in$ {CFS, Chi, Info}, when applied on a $ALL$ data set, selects certain number features. Let us call this as "Cycle1". If we remove all the features selected in "Cycle1" from $ALL$, we get a sub data set, say $ALL1$. Again, we can apply the same feature selection method (applied in Cycle1) on $ALL1$ to see any further features being selected. We call this as "Cycle2". This process stops when no further features are selected.

**Committee of Features :** We have performance by $S(M, 2)$ on the 3 problems with features selected in "Cycle1" in Sections 4.5.1, 4.5.2, and 4.5.3 on 5 functions of yeast, 3 types of protein sites and 26 functions of yeast, respectively. Now we combine the features of "Cycle2" with that of "Cycle1" and monitor the performances through different algorithms.

In Table 5.39 we show the performance by $S(M, 2)$ on selected features through CFS method at "Cycle1" and at "Cycle2" on 3 types of protein sites and performances from Hill method through different algorithms. To note that "Cycle2" consists of all the new features selected at "Cycle1 and Cycle2" for the model building purposes.

The table clearly shows that the concept of committee of features being selected recursively and augmented in models has some improvements for $SVM$ and $NBay$

Table 5.39: Performance by $S(M, 2)$ on 3 types of protein sites by CFS at Cycle1, Cycle2, and Hill through C4.5, NBay, SVM, and MLP.

| Sites | Algorithms | CYCLE1 | CYCLE2 | Hill |
|---|---|---|---|---|
| CALCIUM | C4.5 | 184 | | 185 |
| | SVM | 153 | 157 | 169 |
| | NBay | 144 | 145 | 157 |
| | MLP | 182 | | 183 |
| SERINE | C4.5 | 83 | | 86 |
| | SVM | 22 | 76 | 76 |
| | NBay | 78 | 80 | 79 |
| | MLP | 85 | | 85 |
| DISULFIDE | C4.5 | 67 | | 68 |
| | SVM | 28 | 50 | 64 |
| | NBay | 25 | 43 | 53 |
| | MLP | 36 | 68 | 65 |

on 3 types of protein sites. However we also shown that this process does not further improve performance by $S(M, 2)$ on all the 3 types of protein sites and also always yield lesser performances when compared to the same by Hill method.

In Table 5.40 we show the performance by $S(M, 2)$ on selected features through CFS method at "Cycle1" and at "Cycle2" on 24 functions of yeast (two functions 14.07 and 42.01, could not achieve any improvement by any method are omitted) and performances from Greedy-Hill method through different algorithms. To note that "Cycle2" consists of all the new features selected at "Cycle1 and Cycle2" for the model building purposes.

The above tables shows that number of functions which got improvement in performance by $S(M, 2)$ by different algorithms are—NBay:1 C4.5:8, MLP:9, and SVM:4—out of 24 functions. Here committee of feature has some marginal improvements, but lesser than when compared to Greedy-Hill outcomes.

Table 5.40: Performance by $S(M, 2)$ on 24 functions of yeast by CFS at Cyc1:Cycle1, Cyc2:Cycle2, and Hill through C4.5, NBay, SVM, and MLP.

| 26-func | Code | Genes | NBay | | C4.5 | | MLP | | SVM | | Greedy-Hill |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Cyc1 | Cyc2 | Cyc1 | Cyc2 | Cyc1 | Cyc2 | Cyc1 | Cyc2 | (SVM) |
| 11.02 | Rsn | 226 | -390 | -512 | -76 | -75 | -20 | -42 | -10 | -56 | 16 |
| 11.04 | Rpr | 161 | -192 | -210 | -27 | -59 | -19 | -12 | -28 | -60 | 0 |
| 10.03 | Cyc | 149 | -227 | -337 | -60 | -53 | -40 | -39 | -8 | -42 | 11 |
| 20.09 | Trt | 145 | -452 | -576 | -28 | -28 | -53 | -61 | -1 | -11 | 2 |
| 12.01 | Rib | 138 | 184 | 167 | 203 | 199 | 223 | 225 | 223 | 197 | 239 |
| 1.01 | Aam | 103 | 2 | -20 | 42 | 41 | 55 | 71 | 56 | 66 | 59 |
| 1.06 | Lim | 99 | -49 | -87 | -1 | 4 | -15 | 4 | 10 | 5 | 0 |
| 10.01 | Dna | 99 | -196 | -317 | -30 | -37 | -16 | -22 | -1 | -30 | 3 |
| 1.05 | Ccm | 82 | -135 | -218 | -35 | -45 | -30 | -40 | -6 | -44 | 2 |
| 1.03 | Nuc | 81 | -79 | -124 | -10 | -14 | 4 | -3 | 3 | 6 | 14 |
| 14.13 | Deg | 77 | -167 | -222 | -12 | -5 | 13 | 11 | 11 | -5 | 0 |
| 32.01 | Str | 58 | -110 | -133 | -8 | -1 | 0 | 6 | -1 | 7 | 7 |
| 1.07 | Vit | 54 | -87 | -183 | 0 | 0 | -5 | -4 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | -57 | -149 | 4 | -3 | 1 | -15 | 0 | -2 | 0 |
| 12.04 | Tra | 42 | -199 | -269 | -22 | -27 | -15 | -15 | -8 | -36 | 0 |
| 11 | Tcp | 39 | -3 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 | 0 | -2 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | -75 | -93 | 5 | 8 | 8 | 9 | 7 | 7 | 15 |
| 20.03 | Tfc | 32 | -28 | -39 | 0 | -5 | 0 | -2 | 0 | 0 | 0 |
| 12.10 | Ami | 31 | -48 | -92 | 2 | 6 | 2 | -1 | 0 | 1 | 5 |
| 43.01 | Fun | 31 | -9 | -46 | 0 | -4 | 0 | -11 | 0 | 0 | 0 |
| 2.13 | Res | 29 | -78 | -91 | 5 | 2 | 11 | 8 | 8 | 5 | 11 |
| 14.01 | Pfs | 29 | -11 | -18 | -7 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32.07 | Dtx | 27 | -34 | -67 | -6 | -6 | -8 | -6 | 0 | 0 | 8 |

Other feature selection methods—Chi-square, Information-gain evaluated through committee of features could not improve on any of the above problems. On 3 types of protein sites, we observe performance by $S(M, 2)$ for SERINE by NBay-Cycle2:80 against NBay-Greedy-Hill:79 and for DISULFIDE by MLP-Cycle2:68 against MLP-Greedy-Hill:65. Similarly on 24 function of yeast, we observe for function 1.01 by MLP-Cycle2:71 against SVM-Greedy-Hill:59 and for 1.06 by MLP-Cycle2:4 against SVM-Greedy-Hill:0. This tells us that some features are missed out while Greedy-Hill achieves the best performances.

### 5.8.3 Committee Method

In previous subsection we use only $S(M, 2)$ measure to validate performances using committee of features from "Cycle1" and "Cycle2" on problems. In this subsection we discuss on committee method.

**Sets of Features :** We use conventional feature selection methods—CFS, Chi, and Info (at "Cycle1" as described in earlier subsection) on a data set. We get feature sets (say, $CFS$, $Chi$, and $Info$) for each functions. We have 6 data sets for 5 functions of yeast (say, $D1, D2, D3, D4, D5, andD6$). Also, Hill chooses data sets for each function (say, $Hill$). Now, we have a collection of feature space $FSET \in \{CFS, Chi, Info, D1, D2, D3, D4, D5, D6, Hill\}$. Some researchers use "Committee Method" as combining algorithms—C4.5 and Boosting or Bagging. But we use it differently. Also we found using such method does not increase performance by $S(M, 2)$ on 5 functions of yeast genes.

Now, we use Hill method on $FSET$ for 5 functions of yeast to choose the best combination by learning algorithms with higher performances, for each function. We show in Table 5.41 performances through committee method (only cases where performances has change), Hill, and Exhaustive search (EXH).

Table 5.41: Performance by $S(M, 2)$ on 5 functions of yeast through committee method, Hill, and EXH through C4.5, MLP, and NBay.

| Algorithms | Function | Com-Feat | S(Com) | S(Hill) | S(EXH) |
|---|---|---|---|---|---|
| C4.5 | TCA | Hill-cfs | 3 | 0 | 6 |
| MLP | HIST | D3-cfs | 18 | 16 | 16 |
| MLP | PROT | cfs-Hill | 44 | 39 | 40 |
| MLP | RESP | D5-cfs | 0 | 0 | 2 |
| MLP | RIBO | cfs-chi-Hill | 214 | 230 | 235 |
| MLP | TCA | chi-cfs | 0 | 4 | 12 |
| Nbay | PROT | Hill-cfs | 35 | 1 | 21 |
| Nbay | RIBO | cfs-chi-Hill | 194 | 227 | 228 |

The table shows that committee method achieve higher performances than Hill in 4 out of 8 cases and poorer in 3 out of 8 cases. Similarly committee method achieve

higher performances than EXH in 3 out of 8 cases and poorer in 5 out of 8 cases. The three cases—HIST, MLP:D3-CFS, PROT,MLP:CFS-Hill, PROT,NBay:Hill-CFS confirms that $Hill$ can even surpass $EXH$ on more functions, if useful features from $CFS$ are augmented into the model (as Hill-CFS).

### 5.8.4 18 Function Through Statistical Sampling

In this chapter performances on functions of yeast are illustrated. Only 13 out of 26 function (=50%) and 4 out of 20 functions (=20%) could be modeled (we use a threshold of $S(M, 2) > 0$) by Hill or Greedy-Hill. Number of functions modeled by other approaches are—on 26 function:ALL:4, BI:7, CFS:6, Chi:4, Info:4 and on 20 function:ALL:2, BI:0, CFS:0, Chi:0, Info:0.

In this subsection we summarizes the performance on different approaches on 18 protein functions, which are selected based on sample size and their performances. Column 2 shows whether a function belongs 26 function or 20 function set. Column 3 shows the function code. Column 4 and 5 show the number of positive genes or samples and negative samples, respectively in the data. Column 6 shows the performance $S(ALL + SVM, 2)$ for a protein function $f$ using all available data sets. Column 7 shows the performance $S(Hill + SVM, 2)$ for a protein function $f$ using the combination of whole data sets chosen by Hill. Column 8 shows the performance $S(Greedy\text{-}Hill + SVM, 2)$ for a protein function $f$ using the combination of whole data sets chosen by Greedy-Hill.

Function 11.04 with 161 genes achieve S(M,2)=0 by Hill and Greedy-Hill. This shows that performance by $S(M, 2)$ is not based on number of genes in the data set. Now we illustrate statistical based sampling on choosing sets of negative samples against positive samples in classification modeling.

Table 5.42: Performance by $S(M, 2)$ on 18 functions of yeast through ALL, Hill, Greedy-Hill through SVM.

| Function Catalogue Number | Function Set | Function Code | Number of Positive Samples (genes) | Number of Negative Samples | S(M,2) by ALL | S(M,2) by Hill | S(M,2) by Greedy-Hill |
|---|---|---|---|---|---|---|---|
| 1.01 | 26func | Aam | 103 | 2019 | 46 | 37 | 59 |
| 1.06 | 26func | Lim | 99 | 2023 | -63 | 0 | 0 |
| 20.09 | 26func | Trt | 145 | 1977 | -119 | 2 | 2 |
| 11.04 | 26func | Rpr | 161 | 1961 | -140 | 0 | 0 |
| 14.13 | 26func | Deg | 77 | 2045 | -15 | 0 | 0 |
| 1.07 | 26func | Vit | 54 | 2068 | -34 | 0 | 0 |
| 2.13 | 26func | Res | 29 | 2093 | 3 | 2 | 11 |
| 11.02 | 26func | Rsn | 226 | 1896 | -149 | 16 | 16 |
| 12.01 | 26func | Rib | 138 | 1984 | 228 | 237 | 239 |
| 14.07 | 26func | Prm | 48 | 2074 | -61 | 0 | 0 |
| 20.03 | 26func | Tfc | 32 | 2090 | -23 | 0 | 0 |
| 32.07 | 26func | Dtx | 27 | 2095 | -8 | 4 | 8 |
| 1.02 | 20func | Nsm | 13 | 2109 | 0 | 4 | 4 |
| 2.10 | 20func | Tca | 6 | 2116 | -3 | 0 | 0 |
| 2.16 | 20func | Fer | 7 | 2115 | -1 | 0 | 0 |
| 11.06 | 20func | Rmo | 20 | 2102 | -8 | 0 | 0 |
| 42.04 | 20func | Cyt | 3 | 2119 | 0 | 0 | 0 |
| 42.10 | 20func | Nuc | 5 | 2117 | -1 | 0 | 0 |

In Table 5.43 cross validation folds used in the model are given in the format—Training data:Number of Positive folds/total folds, Number of Negative folds/total folds; Test data:Number of Positive folds/total folds, Number of negative folds/total folds (TR:P2/3,N1/9, TE:p1/3, N1/9)—in column 2. Column 3 to 7 shows performances $S(M, 2)$, Sensitivity, Precision, Accuracy, and Specificity for each set of data for algorithms shown in column 1.

Using 2-3Fold in training and 1-3Fold in testing for positive and negative (this scheme is used in the thesis) achieve higher performances by all algorithms. The study on multiple folds on training and testing over all 18 functions demonstrated that performances achieved by Greedy-Hill are better than conventional methods, statistical sampling, and committee methods. Due to space constraint we could not illustrate outcome on 17 functions by *ALL* and Greedy-Hill by statistical sampling.

Table 5.43: Performance by $S(M, 2)$ on function 11.04 (Positive samples:161 and Negatives samples:1961) using Hill method on different cross validation folds and learning algorithms.

| Algorithm | Number of Folds in a Model | S(M,2) | Sensitivity | Precision | Accuracy | Specificity |
|---|---|---|---|---|---|---|
| C4.5 | TR:P2/3,N1/9,TE:P1/3,N1/9 | -89.00 | 0.226 | 0.184 | 0.865 | 0.918 |
| | TR:P2/3,N1/3,TE:P1/3,N2/3 | 3.00 | 0.112 | 0.353 | 0.917 | 0.983 |
| | TR:P2/3,N1/3,TE:P1/3,N1/3 | 5.00 | 0.112 | 0.365 | 0.918 | 0.984 |
| | TR:P2/3,N2/3,TE:P1/3,N1/3 | 13.00 | 0.093 | 0.469 | 1.008 | 0.991 |
| MLP | TR:P2/3,N1/9,TE:P1/3,N1/9 | -292.00 | 0.478 | 0.147 | 0.750 | 0.772 |
| | TR:P2/3,N1/3,TE:P1/3,N2/3 | -81.00 | 0.267 | 0.205 | 0.866 | 0.915 |
| | TR:P2/3,N1/3,TE:P1/3,N1/3 | -68.00 | 0.269 | 0.220 | 0.872 | 0.921 |
| | TR:P2/3,N2/3,TE:P1/3,N1/3 | 23.00 | 0.112 | 0.581 | 1.009 | 0.993 |
| NBay | TR:P2/3,N1/9,TE:P1/3,N1/9 | -226.00 | 0.432 | 0.160 | 0.785 | 0.814 |
| | TR:P2/3,N1/3,TE:P1/3,N2/3 | -25.00 | 0.199 | 0.264 | 0.897 | 0.955 |
| | TR:P2/3,N1/3,TE:P1/3,N1/3 | -23.00 | 0.203 | 0.268 | 0.897 | 0.954 |
| | TR:P2/3,N2/3,TE:P1/3,N1/3 | 8.00 | 0.143 | 0.377 | 1.012 | 0.981 |
| SVM | TR:P2/3,N1/9,TE:P1/3,N1/9 | -84.00 | 0.169 | 0.164 | 0.872 | 0.929 |
| | TR:P2/3,N1/3,TE:P1/3,N2/3 | -2.00 | 0.000 | 0.000 | 0.923 | 0.999 |
| | TR:P2/3,N1/3,TE:P1/3,N1/3 | -1.00 | 0.000 | 0.000 | 0.924 | 0.999 |
| | TR:P2/3,N2/3,TE:P1/3,N1/3 | 0.00 | 0.000 | 0.000 | 1.000 | 1.000 |

# Chapter 6

# Conclusions

Previous researchers on the three biological problems considered in this thesis (Brown *et al.*, Mateos *et al.*, Bagley *et al.* and Wei *et al.*) use all available data sets together as one single combined data set. They did not show whether using all available data sets would consistently lead to a better performance by $S(M, 2)$ than using a judiciously chosen smaller combination of data sets. They also did not investigate the issue of the optimal choice of combinations of data sets. In this chapter we list our contributions, illustrate the challenges addressed and finally suggest future directions.

- We introduced the Progressive Data Mining (PDM) concept based on whole-dataset feature selection.
- We designed the Hill climbing algorithm (Hill) to perform whole-dataset feature selection. Hill handles a small number of data sets to achieve better classifiers.
- We further designed the Greedy-Hill climbing algorithm (Greedy-Hill). Greedy-Hill is an improvement to Hill as it is able to handle a much larger number of data sets to achieve better classifiers.
- We demonstrated that PDM through Hill achieved a better performance than

conventional methods and equal performance by $S(M, 2)$ to "Exhaustive search" for the purposes of predicting :

1. 5 specific protein functions of yeast genes
2. 3 types of specific protein sites

- We also showed that PDM through Greedy-Hill achieved a better performance by $S(M, 2)$ than conventional methods for the purpose of predicting

  1. 26 specific protein functions of yeast genes

- We showed that PDM selected useful data sets, thereby giving directions to biologists on important experiments for their study. The following questions were addressed :

  1. Can we use limited biological samples to build a proper classifier? We used a small number of samples on the 3 bioinformatics prediction problems—5 specific protein functions of yeast genes, 3 types of protein sites, and 26 specific functions of yeast genes, and proved that we could build good classification models through limited biological samples.
  2. Can we use additional data sets, on the same set of genes or sies with different experimental nature, to improve a classifier? We demonstrated in Chapter 4 that judicious use of additional data sets—even those that were derived from very different wet experimental conditions (different sets of features on same set of genes or sites)—could increase the accuracy of classification models on the 3 bioinformatics prediction problems.

3. Can we use all available data sets to achieve a better classifier? We cautioned in Chapter 4, that using all available data sets did not give the best improved prediction accuracy, and often gave a worse accuracy on the 3 bioinformatics prediction problems.

4. Can we use selected features from conventional feature selection method to achieve a better classifier? We illustrated in Chapter 4 how prediction accuracy could be improved by using conventional feature selection methods compared to using the best individual data sets or all available data sets. However, we also showed that conventional feature selection methods did not achieve the best prediction accuracy often enough than using a combination of whole individual data sets.

5. Can we combine selected data sets to yield a better classifier? We introduced Progressive Data Mining through Hill climbing method (*Hill* that handles a small number of data sets) and Greedy-Hill climbing method (*Greedy-Hill* that handles a larger number of data sets) to achieve better classifiers. We illustrated in Chapter 5 that Hill achieved a better performance by $S(M, 2)$ than conventional methods for the purpose of predicting 5 specific protein functions of yeast genes and 3 types of specific protein sites and Greedy-Hill on 26 specific protein functions of yeast genes. We further validated that Hill and Greedy-Hill produced results that were very close to optimum compared to exhaustive search.

**Research dimensions, possible future directions :**

Our new idea on "selection of important data sets for obtaining better decision making", is an important focus of many industry. Decision making is a key to success

of any organisation; and the tools required for it is very important. Our research is opening many dimensions not only in the biological domain, but also in other domains like—Banking, Insurance, Health-care etc. On the other hand when more and more data flows into the decision space, it is really difficult to decide how to control or manage voluminous of data.

- Applying graph theory techniques to partition input data sets and transpose them into groups—useful and not-useful—on to a 2-dimensional map for selection of data sets. Apply distance measures to formulate phylogeny tree and select data sets.

- Apply Operation Research techniques (multivariate, linear programming, dynamic programming) to evaluate data sets and selects it for better classification models.

- A Web Based System ***VINESIAN*** "Varieties of Information in N-dimension is Evaluated Systematically and Important sets Are Notified". The system takes input of many odd types of data—Sequences, Abstracts, Protein interactions, Gene locations, Microarray profiles etc and build a uniform data structure for genes. A auto updating of annotations from latest catalogue pertaining to the organism of interest is done for the genes tabulated. Then the system integrate with other existing data mining softwares like WEKA [65], GIST [5] and build classification models. It also gives many comparison of methods, approaches through—graphs, tables etc.

- Validation of different data types—Microarray profiles, Higher order features

from Emerging Patterns, Abstract as additional feature space, sequences, protein-protein interactions, Gene location–to be evaluated by "Hill" or "Greedy-Hill" for better classification models on Yeast.

- "Hill" and "Greedy-Hill" with "Whole Dataset Feature Selection Method" to be applied for multi positive classes against negative classes [2-class to n-classes] modeling.

- "Hill" and "Greedy-Hill" can be used to select important data sets on other organism—Homo sapiens, Mus musculus, and Rattus norvegicus from SMD (http://genome-www5.stanford.edu/cgi-bin/source/sourceSearch)—Standford Microarray Data bases depository.

- Gene functions can be structured by functional hierarchy to build more hierarchical classifiers and predictors.

- Protein sites : Disulfide bridging patterns as additional feature to be evaluated for better model for Disulfide bridging of known and used for predicting unknowns.

- Banking and Insurance

  The focus of this research is on showing the use of heterogeneous data sets by whole dataset feature selection on biological data sets. So, we do not discuss much on the sectors of Banking and Insurance in our thesis report. How-ever, one can take this as a future direction, to answer:

  1. Does limited data help in decision making?
  2. Does additional data help in better decision making?

3. Does using all available data give a best decision?

4. Does applying filtering method help in better decision?

5. Does choosing important data help efficiently decision making?

# Bibliography


[1] **Alter O, Brown P.O, Botstein D**, Singular value decomposition for genome-wide expression data processing and modeling, *Proc National Academic Science* USA, **97(18)**, (2000) 10101-6.

[2] **Bagirov A.M et al.**, New algorithms for multi-class cancer diagnosis using tumor gene expression signatures. *Bioinformatics*, **19**, 14, (2003) 1800-1807.

[3] **Binkowski, T.A, Naghibzadeh, S, Liang, J**, CASTp: Computed Atlas of Surface Topography of proteins. *Nucleic Acids Research*, **31**, 13, (2003) 3352-3355,

[4] **Banatao D.R, Altman, R.B, Klein, T.E**, Micro-environment analysis and identification of magnesium binding sites in RNA, *Nucleic Acids Research*, **31**, (2003) 4450-4460.

[5] **Brown, M.P, Grundy, W.N, Lin, D**, Knowledge-based analysis of microarray gene expression data by using support vector machines, *Proc National Academic Science* U S A, **97**, (2000) 262-7.

[6] **Cheng, Y, Church, G.M**, Biclustering of expression data, *Proc Int Conf Intell Syst Mol Biol*, **8**, (2000) 93-103.

[7] **Chu, S, DeRisi, J, Eisen, M, and others**, The transcriptional program of sporulation in budding yeast, *Science*, **282**, (1998) 699-705.

[8] **Clare, A, King, R.D**, Predicting gene function in Saccharomyces cerevisiae, *Bioinformatics*, **19**, Suppl.2, (2003) ii42-ii49.

[9] **DeRisi, J.L, Iyer, V.R, Brown, P.O**, Exploring the metabolic and genetic control of gene expression on a genomic scale, *Science*, **278**, (1997) 680-6.

[10] **Eisen, M.B, Spellman, P.T, Brown, P.O, Botstein, D**, Cluster analysis and display of genome-wide expression patterns, *Proc National Academic Science* U S A, **95**, (1998) 14863-8.

[11] **Fernandes P.M, Domitrovic T, Kao, C.M, Kurtenbach, E**, Genomic expression pattern in Saccharomyces cerevisiae cells in response to high hydrostatic pressure, *FEBS Lett*, **556(1-3)**, (2004) 153-60.

[12] **Gardiner-Garden, M, Little john, T.G**, A comparison of microarray databases, *Brief Bioinform*, **2**, (2001) 143-58.

[13] **Gasch, A.P, Spellman, P.T, Kao, C.M, Carmel-Harel, O, Eisen, M.B, Storz, G, Botstein, D, Brown, P.O**, Genomic expression programs in the response of yeast cells to environmental changes, *Mol Biol Cell*, **11(12)**, (2000) 4241-57.

[14] **Gasch, A.P, Huang, M, Metzner, S, Botstein, D, Elledge, S.J, Brown, P.O**, Genomic expression responses to dna-damaging agents and the regulatory role of the yeast atr homolog mec1p, *Mol Biol Cell*, **12(10)**, (2001) 2987-3003.

[15] **George, H, John, Pat Langley**, Estimating Continuous Distributions in Bayesian Classifiers. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*. **Morgan Kaufmann**, San Mateo, (1995) 338-345.

[16] **Gollub, J, Ball, C.A, Binkley, G, and others**, The Stanford Microarray Database: data access and quality assessment tools, *Nucleic Acids Res*, **31**, (2003) 94-6.

[17] **Golub et al**, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, (1999) 531-537.

[18] **Gross, C, Kelleher, M, Iyer, V.R, Brown, P.O, Winge, D.R**, Identification of the copper regulon in Saccharomyces cerevisiae by DNA microarrays, *J Biol Chem*, **275(41)**, (2000) 32310-6.

[19] **Hall, M.A**, Correlation-based Feature Subset Selection for Machine Learning. Thesis submitted in partial fulfillment of the requirements of the degree of Doctor of Philosophy at the University of Waikato, 1998.

[20] **Huberman, J.A**, Cell cycle control of S phase: a comparison of two yeasts, *Chromosoma*, **105**, 4, (1996) 197-203.

[21] **Hvidsten, T.R, Komorowski, J, Sandvik, A.K, and Laegreid, A**, Predicting gene function from gene expressions and ontologies, *Pac Symp Biocomput*, (2001) 299-310.

[22] **Keller, G, Ray, E, Brown, P.O, Winge, D.R**, Haa1 a protein homologous to the copper-regulated transcription factor Ace1, is a novel transcriptional activator, *J Biol Chem*, **276(42)**, (2001) 38697-702.

[23] **Keerthi, S.S, Shevade, S.K, Bhattacharyya, C, Murthy, K.R.K**, Improvements to Platt's SMO Algorithm for SVM Classifier Design, *Neural Computation*, **13(3)**, (2001) 637-649.

[24] **Li, J, Liu, H, Downing, J.R, and others**, Simple rules underlying gene expression profiles of more than six subtypes of acute lymphoblastic leukemia (ALL) patients, *Bioinformatics*, **19** (2003) 71-8.

[25] **Liang, J, Edelsbrunner, H, Fu, P, Sudhakar, P.V, Subramaniam, S**, Analytical shape computing of macromolecules I: molecular area and volume through alpha shape. *Proteins*, **33**, (1998a), 1-17.

[26] **Liang, J, Edelsbrunner, H, Fu, P, Sudhakar, P.V, Subramaniam, S**, Analytical shape computing of macromolecules II: identification and computation of inaccessible cavities inside proteins. *Proteins*, **33**, (1998b) 18-29.

[27] **Liang, J, Edelsbrunner, H, Woodward, C**, Anatomy of Protein Pockets and Cavities: Measurement of Binding Site Geometry and Implications for Ligand Design. *Protein Science*, **7**, (1998c) 1884-1897.

[28] **Lill, R, Nargang, F.E, Neupert, W**, Biogenesis of mitochondrial proteins, *Curr Opin Cell Biol* **8**, 4, (1996) 505-12.

[29] **Wei, L, Altman, R.B**, Recognizing complex, asymmetric functional sites in protein structure using a Bayesian scoring function, *Journal of bioinformatics and computational biology*, **1**, (2003) 119-138.

[30] **Wei, L, Enoch, S.H, Altman, R.B**, Are predicted structures good enough to preserve functional sites?, *Structure*, **7**, (1999) 643-650.

[31] **Wei, L, Altman, R.B**, Recognizing protein binding sites using statistical descriptions of their 3D environments. *Pacific Symposium on Bioinformatics*, 1998.

[32] **Liu, H, Setiono, R**, $\chi^2$: Feature selection and discretization of numeric attributes. *In Proceedings of the IEEE 7th International Conference on Tools with Artificial Intelligence*, (1995) 338-391.

[33] **Liu, H, Li, J, Wong, L**, A comparative study of feature selection and classification methods using gene expression profiles and proteomic patterns, *Genome Informatics*, **13**, (2002) 51-60.

[34] **Mateos, A, Dopazo, J, Jansen, R, and others**, Systematic learning of gene functional classes from DNA array expression data by using multilayer perceptrons, *Genome Research*, **12**, (2002) 1703-15.

[35] **Medvedovic, M, Sivaganesan, S**, Bayesian infinite mixture model based clustering of gene expression profiles, *Bioinformatics*, **18**, 9, (2002) 1194-206.

[36] **Mewes, H.W, Frishman, D, Guldener, U, and others**, MIPS: a database for genomes and protein sequences, *Nucleic Acids Res*, **30**, (2002) 31-4.

[37] **Miller, L.D, Long, P.M, Wong, L, and others**, Optimal gene expression analysis by microarrays, *Cancer Cell*, **2**, (2002), 353-61.

[38] **Molina, M, Sanchez, M.H, Nombela, C.**, MAP kinase-mediated signal transduction pathways, *Yeast Gene Analysis*, **Tuite, M.F, Brown, P.J.P**, *Methods in Microbiology*, Acad. Press, 26, 1998.

[39] **Nayal, M, Di Cer, E**, $Ca^{2+}-$binding sites in proteins, *Proc Natl. Acad. Sci.*, **91**, (1994) 817-821.

[40] **Ogawa, N, DeRisi, J, Brown, P.O**, New components of a system for phosphate accumulation and polyphosphate metabolism in Saccharomyces cerevisiae revealed by genomic expression analysis, *Mol Biol Cell*, **11**, (2000) 4309-21.

[41] **O'Neill, M.C, Song, L**, Neural Network Analysis of Lymphoma Microarray Data: Prognosis and Diagnosis Near-Perfect, *BMC Bioinformatics*, **4**, (2003) 1,13.

[42] **Protchenko, O, Ferea, T, Rashford, J, Tiedeman, J, Brown, P.O, Botstein, D, Philpott, C.C**, Three cell wall mannoproteins facilitate the uptake of iron in Saccharomyces cerevisiae, *J Biol Chem*, **276(52)**, (2001) 49244-50.

[43] **J. Platt**. Fast Training of Support Vector Machines using Sequential Minimal Optimization. Advances in Kernel Methods - Support Vector Learning, Schoelkopf, B, Burges, C, Smola, A, eds, *MIT Press* (1998).

[44] **Philippe**. Extracting pathways from gene expression data, *Bioinformatics*, **19 supl2**, 2003.

[45] **Ramaswamy, S et al.**, Multi-class cancer diagnosis using tumor gene expression signatures, *Proc. National Academic Science*, **98**, (2001) 15149-15154.

[46] **Chen-Hsiang Yeang et al.**, Molecular classification of multiple tumor types, *Bioinformatics*, **17**, (2001) S316-S322.

[47] **Reggiori, F, Conzelmann, A**, Biosynthesis of inositol phosphoceramides and remodeling of glycosylphosphatidylinositol anchors in Saccharomyces cerevisiae are mediated by different enzymes, *J Biol Chem*, **273**, 46, (1998) 30550-9.

[48] **Rutherford J.C, Jaron S, Ray, E, Brown P.O, Winge, D.R**, A second iron-regulatory system in yeast independent of Aft1p, *Proc National Academic Science*, U S A **98(25)**, (2001) 14322-7.

[49] **Sawa, T, Ohno-Machado, L**, A neural network-based similarity index for clustering DNA microarray data,*Comput Biol Med*, **33**, 1, (2003) 1-15.

[50] **Schulze, A, Downward, J**, Navigating gene expression using microarrays: a technology review, *Nat Cell Biol*, **3**, 8, (2001) E190-5.

[51] **See-Kiong Ng, Soon Heng Tan, Sundararajan, V.S**, On combining Multiple Microarray Studies for Improved Functional Classification by Whole-Dataset Feature Selection, GIW2003, *Genome Informatics* **14**, (2003) 44-53.

[52] **Segal, E, Shapira, M, Regev, A, Peer, D, Botstein, D, Koller, D, Friedman, N**, Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data, *Nat Genet.*, **34(2)**, (2003) 166-176.

[53] **Shakoury-Elizeh, M, Tiedeman, J, Rashford, J, Ferea, T, Demeter, J, Garcia, E, Rolfes, R, Brown, P.O, Botstein, D, Philpott, CC**, Transcriptional remodeling in response to Iron deprivation in *Saccharomyces cerevisiae*, *Mol. Biol. Cell*, **15(3)** (2004) 1233-43.

[54] **Shannon, W, Culverhouse, R, Duncan, J**, Analyzing microarray data using cluster analysis, *Pharmacogenomics*, **4**, 1, (2003) 41-52.

[55] **Sotriffer, C, Klebe, G**, Identification and mapping of small-molecule binding sites in proteins:computational tools for structure-based drug design, *IL Farmaco*, **57**, (2002) 243-251.

[56] **Spellman, P.T, Sherlock, G, Zhang, M.Q, and others**, Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization, *Mol Biol Cell*, **9**, (1998) 3273-97.

[57] **Steven, C, Bagley, Russ B, Altman**, Characterizing the micro-environment surrounding protein sites. *Protein Science*, **4**, (1995) 622-635.

[58] **Steven C, Bagley, Liping Wei, Russ B, Altman**, Characterizing oriented protein structural sites using biochemical properties. *International conference on intelligent systems for Mol. Biol.*, (1995) 12-20.

[59] **Sudarsanam, P, Iyer, V.R, Brown, P.O, Winston, F**, Whole-genome expression analysis of snf/swi mutants of Saccharomycescerevisiae, *Proc National Academic Science*, U S A, **97**, (2000) 3364-9.

[60] **Theilhaber, J, Connolly, T, Roman-Roman, S, and others**, Finding genes in the C2C12 osteogenic pathway by k-nearest-neighbor classification of expression data, *Genome Res*, **12**, 1, (2002) 165-76.

[61] **Thomas, D, Sudrdin-Kerjan, Y**, *Microbiology and Molecular Biology Reviews*, **61**, (1997) 503-532.

[62] **Vert, J.P, Kaneshisa, M**, Extracting active pathways from gene expression data, *Bioinformatics*, **19**, Suppl.2, (2003) ii238-ii244.

[63] **Wagner, R, de Montigny, J, de Wergifosse, P, and others**, The ORF YBL042 of Saccharomyces cerevisiae encodes a uridine permease *FEMS Microbiol Lett*, **159**, 1, (1998) 69-75.

[64] **Walker, M.G, Volkmuth, W, Sprinzak, E, Hodgson, D, Klingler, T**, Prediction of Gene Function by Genome-Scale Expression Analysis: Prostate Cancer-Associated Genes, *Genome Research*, **9**, 12, (1999) 1198-1203.

[65] **Witten, I.H, Frank, E**, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, 1999.

[66] **Qunlan J.R**, C4.5: Programs for Machine Learning, *Morgan Kaufmann Publishers*, San Mateo, CA. 1993.

[67] **Yoshimoto, H, Saltsman, K, Gasch, A.P, Li HX, Ogawa, N, Botstein, D, Brown, P.O, Cyert, M.S**, Genome-wide Analysis of Gene Expression Regulated by the Calcineurin/Crz1p Signaling Pathway in Saccharomyces cerevisiae, *J Biol Chem*, **277(34)**, (2002) 31079-31088.

[68] **Yamashita, M.M, Wesson, L, Eisenman, G, Eisenbert, D**, (1990) Where metal ions bind in proteins, *Proc Natl. Sci.*, **87**, (1990) 5648-5652.

[69] **Yoshimoto, H, Saltsman, K, Gasch, A.P, and others**, Genome-wide analysis of gene expression regulated by the calcineurin/Crz1p signaling pathway in Saccharomyces cerevisiae, *J Biol Chem*, **277**,34, (2002) 31079-88.

[70] **Zhu, G, Spellman, P.T, Volpe, T, Brown, P.O, Botstein, D, Davis, T.N, Futcher, B**,Two yeast forkhead genes regulate the cell cycle and pseudohyphal growth, *Nature*, **406(6791)**, (2000) 90-4.

[71] An Introduction to Support Vector Machines (www.mathcs.carleton.edu/ faculty/dmusican/).

[72] (www.orsoc.org.uk/conf/previous/yor12/) An Introduction to Support Vector Machines for Data Mining .

[73] (www.cs.ucsd.edu /dboswell/PastWork/) Introduction to Support Vector Machines.

[74] (www.doc.ic.ac.uk/xh1/Referece/) Machine Learning.

[75] (www.ai.rug.nl/ki2/AdvisedReading/ecoe554-10.pdf) Machine Learning.

[76] (www.cis.temple.edu/ingargio/cis587/readings/id3-c45.html) Building Classification Models: ID3 and C4.5.

[77] (www4.cs.umanitoba.ca/jacky/Teaching/Courses/74.436/) Naïve Bayesian Learning.

[78] (www.ai.ijs.si/Mezi/pedagosko/) Implementation of Naïve Bayesian Classifiers in Java.

[79] (www.gatsby.ucl.ac.uk/zoubin/bayesian.html) Bayesian Machine Learning.

[80] Machine Learning (www.cs.wisc.edu/dyer/cs540/notes/learning.html).

[81] (www.eng.auburn.edu/users/gvdozier/ML.ppt) An Introduction to Machine Learning.

[82] Neural Networks (www.doc.ic.ac.uk/nd/surprise_96/journal/vol4/cs11/).

[83] (www.statsoftinc.com/textbook/stneunet.html) Neural Networks.

[84] (www.statsoftinc.com/textbook/glosn.html) Neural Networks .

# Appendix A

# Additional Tables on 5 Functions of Yeast Genes

The combination of Exhaustive search combinations which are Less than, equal to, and greater than the performance from GreedyHill on 5 functions of Yeast.

Table A.1: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 5 protein functions of yeast through NBay.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| HIST | 249 | 4 | 2 | 97.65 | 1.57 | 0.78 |
| PROT | 227 | 2 | 26 | 89.02 | 0.78 | 10.20 |
| RESP | 252 | 3 | 0 | 98.82 | 1.18 | 0.00 |
| RIBO | 252 | 2 | 1 | 98.82 | 0.78 | 0.39 |
| TCA | 254 | 1 | 0 | 99.61 | 0.39 | 0.00 |

Table A.2: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 5 protein functions of yeast through C4.5.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| HIST | 248 | 6 | 1 | 97.25 | 2.35 | 0.39 |
| PROT | 252 | 1 | 2 | 98.82 | 0.39 | 0.78 |
| RESP | 137 | 118 | 0 | 53.73 | 46.27 | 0.00 |
| RIBO | 242 | 3 | 10 | 94.90 | 1.18 | 3.92 |
| TCA | 230 | 14 | 11 | 90.20 | 5.49 | 4.31 |

Table A.3: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 5 protein functions of yeast through MLP.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| HIST | 236 | 19 | 0 | 92.55 | 7.45 | 0.00 |
| PROT | 253 | 1 | 1 | 99.22 | 0.39 | 0.39 |
| RESP | 239 | 15 | 1 | 93.73 | 5.88 | 0.39 |
| RIBO | 189 | 14 | 52 | 74.12 | 5.49 | 20.39 |
| TCA | 239 | 6 | 10 | 93.73 | 2.35 | 3.92 |

Table A.4: Average performances over 5 functions of yeast by Multiple evaluation metrics through C4.5, NBay, and MLP.

| Data | Algorithm | SM | Sensitivity | Precision | FM | Specificty | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|---|
| BI | C4.5 | 40 | 0.387 | 0.427 | 0.406 | 0.997 | 0.989 | 0.607 | 0.003 |
| ALL | C4.5 | 46 | 0.499 | 0.463 | 0.446 | 0.997 | 0.990 | 0.497 | 0.004 |
| CFS | C4.5 | 49 | 0.494 | 0.568 | 0.527 | 0.998 | 0.991 | 0.500 | 0.002 |
| Chi | C4.5 | 51 | 0.552 | 0.589 | 0.567 | 0.997 | 0.992 | 0.444 | 0.003 |
| Info | C4.5 | 51 | 0.552 | 0.589 | 0.567 | 0.997 | 0.992 | 0.444 | 0.003 |
| HILL | C4.5 | 53 | 0.471 | 0.478 | 0.474 | 0.998 | 0.992 | 0.525 | 0.002 |
| Greedy-Hill | C4.5 | 50 | 0.481 | 0.456 | 0.468 | 0.997 | 0.991 | 0.515 | 0.003 |
| EXH | C4.5 | 53 | 0.550 | 0.579 | 0.564 | 0.997 | 0.992 | 0.446 | 0.003 |
| BI | MLP | 48 | 0.476 | 0.608 | 0.519 | 0.998 | 0.991 | 0.520 | 0.002 |
| ALL | MLP | 48 | 0.556 | 0.600 | 0.572 | 0.995 | 0.990 | 0.440 | 0.005 |
| CFS | MLP | 51 | 0.482 | 0.515 | 0.493 | 0.997 | 0.992 | 0.510 | 0.003 |
| Chi | MLP | 55 | 0.618 | 0.607 | 0.607 | 0.996 | 0.992 | 0.378 | 0.004 |
| Info | MLP | 56 | 0.618 | 0.614 | 0.611 | 0.996 | 0.992 | 0.378 | 0.004 |
| HILL | MLP | 58 | 0.523 | 0.653 | 0.557 | 0.998 | 0.994 | 0.473 | 0.002 |
| Greedy-Hill | MLP | 57 | 0.522 | 0.650 | 0.554 | 0.998 | 0.993 | 0.475 | 0.002 |
| EXH | MLP | 61 | 0.596 | 0.842 | 0.628 | 0.998 | 0.994 | 0.400 | 0.002 |
| BI | NBay | 38 | 0.319 | 0.500 | 0.343 | 0.999 | 0.990 | 0.677 | 0.001 |
| ALL | NBay | 15 | 0.749 | 0.489 | 0.526 | 0.977 | 0.974 | 0.247 | 0.025 |
| CFS | NBay | 40 | 0.670 | 0.460 | 0.530 | 0.988 | 0.985 | 0.327 | 0.012 |
| Chi | NBay | 18 | 0.723 | 0.467 | 0.533 | 0.978 | 0.975 | 0.273 | 0.023 |
| Info | NBay | 18 | 0.723 | 0.467 | 0.533 | 0.978 | 0.975 | 0.273 | 0.023 |
| HILL | NBay | 49 | 0.387 | 0.511 | 0.385 | 0.998 | 0.992 | 0.611 | 0.002 |
| Greedy-Hill | NBay | 48 | 0.366 | 0.523 | 0.380 | 0.998 | 0.992 | 0.631 | 0.002 |
| EXH | NBay | 53 | 0.534 | 0.502 | 0.489 | 0.996 | 0.991 | 0.464 | 0.004 |

# Appendix B

# Additional Tables on 3 Types Protein Sites

The combination of Exhaustive search combinations which are Less than, equal to, and greater than the performance from Greedy-Hill on 3 types of protein sites.

Table B.1: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 3 types of protein sites through NBay.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| DISULFIDE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |

Table B.2: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 3 types of protein sites through C4.5.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 60 | 2 | 1 | 95.238 | 3.175 | 1.587 |
| DISULFIDE | 52 | 2 | 9 | 82.540 | 3.175 | 14.286 |

Table B.3: Number and percentage for EXH<Greedy-Hill, EXH=Greedy-Hill, and EXH>Greedy-Hill on 3 types of protein sites through MLP.

| Protein Function | Les. Greedy-Hill | Eq. Greedy-Hill | Grt. Greedy-Hill | Les. Greedy-Hill% | Eq. Greedy-Hill% | Grt. Greedy-Hill% |
|---|---|---|---|---|---|---|
| CALCIUM | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| SERINE | 62 | 1 | 0 | 98.413 | 1.587 | 0.000 |
| DISULFIDE | 58 | 4 | 1 | 92.063 | 6.349 | 1.587 |

Table B.4: Average of Multiple evaluation metrics over 3 types of protein sites through C4.5, NBay, and MLP.

| Data | Algorithm | SM | Sensitivity | Precision | FM | Specificty | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|---|
| BI | C4.5 | 106 | 0.946 | 0.898 | 0.920 | 0.945 | 0.956 | 0.054 | 0.058 |
| ALL | C4.5 | 110 | 0.969 | 0.949 | 0.959 | 0.975 | 0.977 | 0.031 | 0.026 |
| CFS | C4.5 | 111 | 0.973 | 0.962 | 0.967 | 0.981 | 0.983 | 0.027 | 0.019 |
| Chi | C4.5 | 110 | 0.969 | 0.941 | 0.955 | 0.972 | 0.975 | 0.031 | 0.028 |
| Info | C4.5 | 110 | 0.969 | 0.941 | 0.955 | 0.972 | 0.975 | 0.031 | 0.028 |
| HILL | C4.5 | 113 | 0.982 | 0.978 | 0.980 | 0.991 | 0.990 | 0.018 | 0.009 |
| Greedy-Hill | C4.5 | 110 | 0.980 | 0.946 | 0.963 | 0.981 | 0.983 | 0.018 | 0.020 |
| EXH | C4.5 | 115 | 1.000 | 0.988 | 0.994 | 0.993 | 0.996 | 0.000 | 0.007 |
| BI | MLP | 105 | 0.930 | 0.955 | 0.939 | 0.981 | 0.963 | 0.070 | 0.020 |
| ALL | MLP | 102 | 0.928 | 0.912 | 0.918 | 0.968 | 0.950 | 0.072 | 0.033 |
| CFS | MLP | 101 | 0.847 | 0.923 | 0.874 | 0.971 | 0.950 | 0.153 | 0.030 |
| Chi | MLP | 103 | 0.917 | 0.958 | 0.937 | 0.982 | 0.960 | 0.083 | 0.019 |
| Info | MLP | 104 | 0.926 | 0.950 | 0.938 | 0.979 | 0.960 | 0.074 | 0.021 |
| HILL | MLP | 111 | 0.969 | 0.962 | 0.966 | 0.986 | 0.983 | 0.031 | 0.014 |
| Greedy-Hill | MLP | 111 | 0.969 | 0.962 | 0.966 | 0.986 | 0.983 | 0.031 | 0.014 |
| EXH | MLP | 111 | 0.969 | 0.962 | 0.966 | 0.986 | 0.983 | 0.031 | 0.014 |
| BI | NBay | 87 | 0.791 | 0.933 | 0.856 | 0.971 | 0.909 | 0.209 | 0.030 |
| ALL | NBay | 88 | 0.878 | 0.800 | 0.820 | 0.893 | 0.885 | 0.122 | 0.127 |
| CFS | NBay | 82 | 0.862 | 0.746 | 0.790 | 0.839 | 0.854 | 0.138 | 0.209 |
| Chi | NBay | 88 | 0.890 | 0.795 | 0.823 | 0.886 | 0.883 | 0.110 | 0.136 |
| Info | NBay | 88 | 0.890 | 0.795 | 0.823 | 0.886 | 0.883 | 0.110 | 0.136 |
| HILL | NBay | 96 | 0.872 | 0.904 | 0.888 | 0.939 | 0.929 | 0.128 | 0.069 |
| Greedy-Hill | NBay | 96 | 0.872 | 0.904 | 0.888 | 0.939 | 0.929 | 0.128 | 0.069 |
| EXH | NBay | 96 | 0.872 | 0.904 | 0.888 | 0.939 | 0.929 | 0.128 | 0.069 |

# Appendix C

# Additional Tables on 26 Functions of Yeast Genes

Table C.1: Performance by $S(M, 2)$ on 26 protein functions of Yeast using different methods and C4.5.

| Function | Code | Genes | S(ALL) | S(Best-Ind) | S(Hill) | S(Greedy-Hill) | S(CFS) | S(Chi) | S(Info) |
|---|---|---|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | -98 | 5 | 10 | 20 | -76 | -83 | -84 |
| 11.04 | Rpr | 161 | -91 | 8 | 13 | 14 | -27 | -69 | -67 |
| 10.03 | Cyc | 149 | -98 | 4 | 7 | 7 | -60 | -63 | -63 |
| 20.09 | Trt | 145 | -98 | 0 | 0 | 0 | -28 | -53 | -52 |
| 12.01 | Rib | 138 | 188 | 213 | 217 | 221 | 203 | 187 | 187 |
| 1.01 | Aam | 103 | 8 | 30 | 38 | 65 | 42 | 29 | 29 |
| 1.06 | Lim | 99 | -72 | 3 | 11 | 18 | -1 | -28 | -27 |
| 10.01 | Dna | 99 | -59 | 6 | 8 | 17 | -30 | -32 | -35 |
| 1.05 | Ccm | 82 | -65 | 1 | 2 | 2 | -35 | -63 | -63 |
| 1.03 | Nuc | 81 | -54 | 1 | 7 | 13 | -10 | -12 | -12 |
| 14.13 | Deg | 77 | -18 | 0 | 4 | 32 | -12 | -20 | -15 |
| 32.01 | Str | 58 | -9 | 1 | 3 | 3 | -8 | -10 | -10 |
| 1.07 | Vit | 54 | -53 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.07 | Prm | 48 | -34 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | -31 | 0 | 0 | 0 | 4 | 3 | 2 |
| 12.04 | Tra | 42 | -32 | 0 | 0 | 0 | -22 | -29 | -29 |
| 11 | Tcp | 39 | -46 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14.04 | Ptt | 37 | -4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | -15 | 5 | 14 | 15 | 5 | 5 | 6 |
| 20.03 | Tfc | 32 | -30 | 0 | 0 | 0 | 0 | -5 | -5 |
| 42.01 | Wal | 32 | -28 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12.1 | Ami | 31 | -6 | 3 | 10 | 13 | 2 | -2 | -2 |
| 43.01 | Fun | 31 | -21 | 0 | 0 | 0 | 0 | -4 | -4 |
| 2.13 | Res | 29 | 9 | 3 | 16 | 12 | 5 | 4 | 6 |
| 14.01 | Pfs | 29 | -14 | 0 | 0 | 0 | -7 | 0 | 0 |
| 32.07 | Dtx | 27 | -10 | 0 | 0 | 0 | -6 | -6 | -6 |

Table C.2: Performance by $S(M, 2)$ on 26 protein functions of Yeast using different methods and NBay.

| Function | Code | Genes | S(ALL) | S(Best-Ind) | S(Hill) | S(Greedy-Hill) | S(CFS) | S(Chi) | S(Info) |
|---|---|---|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | -593 | 1 | 2 | 3 | -390 | -557 | -557 |
| 11.04 | Rpr | 161 | -338 | 3 | 8 | 8 | -192 | -271 | -271 |
| 10.03 | Cyc | 149 | -665 | 1 | 2 | 2 | -227 | -468 | -468 |
| 20.09 | Trt | 145 | -646 | 2 | 4 | 4 | -452 | -735 | -735 |
| 12.01 | Rib | 138 | 145 | 144 | 170 | 208 | 184 | 145 | 145 |
| 1.01 | Aam | 103 | -82 | 11 | 11 | 11 | 2 | -49 | -49 |
| 1.06 | Lim | 99 | -440 | 0 | 0 | 0 | -49 | -164 | -164 |
| 10.01 | Dna | 99 | -672 | 0 | 4 | 4 | -196 | -459 | -459 |
| 1.05 | Ccm | 82 | -268 | 0 | 0 | 0 | -135 | -288 | -288 |
| 1.03 | Nuc | 81 | -234 | 0 | 0 | 0 | -79 | -159 | -159 |
| 14.13 | Deg | 77 | -405 | 5 | 7 | 7 | -167 | -414 | -414 |
| 32.01 | Str | 58 | -206 | 0 | 1 | 1 | -110 | -172 | -172 |
| 1.07 | Vit | 54 | -663 | 0 | 0 | 0 | -87 | -183 | -183 |
| 14.07 | Prm | 48 | -622 | 0 | 0 | 0 | -7 | -7 | -7 |
| 20.01 | Tcs | 46 | -311 | 0 | 0 | 0 | -57 | -222 | -222 |
| 12.04 | Tra | 42 | -348 | 0 | 1 | 1 | -199 | -363 | -363 |
| 11 | Tcp | 39 | -258 | 0 | 0 | 0 | -3 | -1 | -1 |
| 14.04 | Ptt | 37 | -659 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34.11 | Csr | 33 | -86 | 0 | 0 | 0 | -75 | -102 | -102 |
| 20.03 | Tfc | 32 | -343 | 0 | 0 | 0 | -28 | -66 | -66 |
| 42.01 | Wal | 32 | -212 | 0 | 0 | 0 | -752 | -752 | -752 |
| 12.1 | Ami | 31 | -220 | 0 | 0 | 0 | -48 | -154 | -154 |
| 43.01 | Fun | 31 | -314 | 0 | 0 | 0 | -9 | -78 | -78 |
| 2.13 | Res | 29 | -116 | 0 | 4 | 4 | -78 | -156 | -156 |
| 14.01 | Pfs | 29 | -282 | 0 | 0 | 0 | -11 | -21 | -21 |
| 32.07 | Dtx | 27 | -103 | 0 | 0 | 0 | -34 | -67 | -67 |

Table C.3: Performance by $S(M, 2)$ on 26 protein functions of Yeast using different methods and MLP.

| Function | Code | Genes | S(ALL) | S(Best-Ind) | S(Hill) | S(Greedy-Hill) | S(CFS) | S(Chi) | S(Info) |
|---|---|---|---|---|---|---|---|---|---|
| 11.02 | Rsn | 226 | -24 | 6 | 11 | 7 | -20 | -28 | -33 |
| 11.04 | Rpr | 161 | -24 | 7 | 23 | 18 | -19 | -19 | -30 |
| 10.03 | Cyc | 149 | -25 | 2 | 9 | 9 | -40 | -22 | -21 |
| 20.09 | Trt | 145 | -40 | 0 | 1 | 2 | -53 | -44 | -35 |
| 12.01 | Rib | 138 | 235 | 217 | 234 | 249 | 223 | 226 | 227 |
| 1.01 | Aam | 103 | 71 | 38 | 51 | 74 | 55 | 59 | 63 |
| 1.06 | Lim | 99 | -12 | 4 | 20 | 20 | -15 | 5 | 8 |
| 10.01 | Dna | 99 | -36 | 13 | 20 | 20 | -16 | -31 | -25 |
| 1.05 | Ccm | 82 | -21 | 4 | 4 | 4 | -30 | -26 | -27 |
| 1.03 | Nuc | 81 | 18 | 7 | 21 | 22 | 4 | -6 | 0 |
| 14.13 | Deg | 77 | 22 | 10 | 28 | 38 | 13 | 7 | 13 |
| 32.01 | Str | 58 | 15 | 2 | 4 | 8 | 0 | -1 | 0 |
| 1.07 | Vit | 54 | -21 | 0 | 0 | 0 | -5 | -4 | -4 |
| 14.07 | Prm | 48 | -11 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.01 | Tcs | 46 | -8 | 2 | 7 | 13 | 1 | -7 | -6 |
| 12.04 | Tra | 42 | -11 | 0 | 4 | 8 | -15 | -14 | -11 |
| 11 | Tcp | 39 | -12 | 0 | 1 | 1 | 0 | 0 | 0 |
| 14.04 | Ptt | 37 | 0 | 0 | 0 | 0 | 0 | -2 | -2 |
| 34.11 | Csr | 33 | 6 | 9 | 15 | 16 | 8 | 11 | 11 |
| 20.03 | Tfc | 32 | -12 | 0 | 0 | 0 | 0 | -1 | -2 |
| 42.01 | Wal | 32 | -15 | 0 | 0 | 0 | -1 | -1 | -1 |
| 12.1 | Ami | 31 | 4 | 4 | 12 | 21 | 2 | 0 | -1 |
| 43.01 | Fun | 31 | -6 | 1 | 1 | 1 | 0 | -6 | -6 |
| 2.13 | Res | 29 | 13 | 8 | 11 | 25 | 11 | 13 | 20 |
| 14.01 | Pfs | 29 | -8 | 0 | 3 | 4 | 0 | 0 | 0 |
| 32.07 | Dtx | 27 | 1 | 4 | 4 | 4 | -8 | -6 | -6 |

Table C.4: Average of Multiple evaluation metrics over 26 specific functions of yeast through C4.5, NBay, and MLP.

| Data | Algorithm | SM | Sensitivity | Precision | FM | Specificity | Accuracy | Rt F N | Rt F P |
|---|---|---|---|---|---|---|---|---|---|
| BI | C4.5 | 11 | 0.063 | 0.274 | 0.085 | 0.999 | 0.967 | 0.937 | 0.001 |
| ALL | C4.5 | -30 | 0.157 | 0.174 | 0.162 | 0.970 | 0.944 | 0.843 | 0.031 |
| CFS | C4.5 | -2 | 0.112 | 0.215 | 0.136 | 0.987 | 0.958 | 0.875 | 0.013 |
| Chi | C4.5 | -10 | 0.128 | 0.187 | 0.147 | 0.982 | 0.954 | 0.859 | 0.019 |
| Info | C4.5 | -9 | 0.127 | 0.190 | 0.147 | 0.982 | 0.954 | 0.860 | 0.019 |
| HILL | C4.5 | 14 | 0.095 | 0.328 | 0.130 | 0.998 | 0.967 | 0.905 | 0.002 |
| Greedy-Hill | C4.5 | 17 | 0.117 | 0.255 | 0.149 | 0.995 | 0.965 | 0.883 | 0.005 |
| BI | MLP | 13 | 0.088 | 0.418 | 0.123 | 0.998 | 0.967 | 0.912 | 0.002 |
| ALL | MLP | 4 | 0.160 | 0.279 | 0.196 | 0.987 | 0.960 | 0.840 | 0.014 |
| CFS | MLP | 4 | 0.135 | 0.251 | 0.165 | 0.988 | 0.960 | 0.852 | 0.013 |
| Chi | MLP | 4 | 0.156 | 0.243 | 0.185 | 0.986 | 0.960 | 0.831 | 0.014 |
| Info | MLP | 5 | 0.165 | 0.251 | 0.194 | 0.986 | 0.960 | 0.822 | 0.014 |
| HILL | MLP | 19 | 0.135 | 0.537 | 0.188 | 0.998 | 0.968 | 0.865 | 0.002 |
| Greedy-Hill | MLP | 22 | 0.149 | 0.257 | 0.182 | 0.989 | 0.961 | 0.851 | 0.012 |
| BI | NBay | 6 | 0.043 | 0.159 | 0.046 | 0.997 | 0.965 | 0.957 | 0.003 |
| ALL | NBay | -344 | 0.512 | 0.098 | 0.155 | 0.787 | 0.782 | 0.488 | 0.312 |
| CFS | NBay | -123 | 0.335 | 0.121 | 0.164 | 0.907 | 0.892 | 0.652 | 0.123 |
| Chi | NBay | -222 | 0.422 | 0.098 | 0.148 | 0.851 | 0.842 | 0.565 | 0.217 |
| Info | NBay | -222 | 0.422 | 0.098 | 0.148 | 0.851 | 0.842 | 0.565 | 0.217 |
| HILL | NBay | 8 | 0.055 | 0.263 | 0.066 | 0.997 | 0.966 | 0.945 | 0.003 |
| Greedy-Hill | NBay | 10 | 0.322 | 0.099 | 0.139 | 0.871 | 0.855 | 0.678 | 0.162 |